On parameterized deformations and unsupervised learning

Hansen, Michael Sass

Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

On parameterized deformations and unsupervised learning

Michael Sass Hansen

Kongens Lyngby 2009
IMM-PHD-2009-70

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

IMM-PHD: ISSN 0909-3192, ISBN 0

Summary

The work presented here consists of contributions in three areas.

An efficient algorithm for calculating the entire regularization path of the support vector domain description (SVDD) is presented. The ability to calculate the entire path with a complexity in the same order as solving the original quadratic problems gives inspiration to utilize the extra information available from the entire path. A method for hierarchical support vector clustering, based on information from the entire regularization path, and multiple Gaussian kernels is described. Bayesian methods are applied in the attempt to draw direct statistical conclusions from the SVDD analysis.

In the context of image registration, different assumptions on the warp fields, namely diffeomorphism and a linear elastic potential in the form of regularization are discussed. A new warp representation which allows statistical analysis on an unrestricted linear parameter space, where all derivatives are defined, is introduced. Furthermore, it is shown that L_2-norm the parameter space introduces a reasonable metric in the actual space of modelled diffeomorphisms. A new parametrization of 3D deformation fields, using potentials and Helmholtz decomposition is also presented. The representation can be considered a natural parametrization for both elastic and fluid image registration due to the decoupling of the parameters. The determinant gradient field is shown to be the first-order small-deformation approximation to the determinant of the Jacobian matrix.

Spline approximations of functions and in particular image registration warp fields are discussed. It is shown how spline bases may be learned from the optimization process, i.e. image registration optimization, and how this may contribute with a reasonable prior, or regularization in the method. A new formula, based on the multivariate di-

vided difference, for explicit calculation of the simplex splines is presented. The formula additionally admits easy calculation of derivatives, both spatial, and with respect to the position of the knots. It is demonstrated that conditions may be set on the knot movements, which ensures that the splines form a partition of unity, even if the knots are not Delaunay. A subdivision scheme is also presented, which requires no recalculation of the configurations of the splines. The use of the splines for image registration is demonstrated, and the inherent smoothing or averaging cost, of selecting warp parameterizations at a specific kernel resolution, has been analyzed. A refinement measure has been derived, which is shown to be efficient for guiding the local mesh layout. With the combination of the refinement measure and the local flexibility of the multivariate B-splines, the warp field is automatically refined in areas where it results in the minimization of the registration cost function.

Resumé

Det præsenterede arbejde består af bidrag inden for tre områder.

Inden for statistiske kernel-metoder er en hurtig algoritme udviklet til beregning af hele regulariseringsstien for support vector domain description - metoden. Metoden er anvendt til at udvikle et framework til hierarkisk support vector clustering, som er baseret på information fra hele regulariseringsstien og multiple Gaussiske kerner. Bayesiske metoder er anvendt i et forsøg på at uddrage direkte statistiske konklusioner fra analysen.

Inden for billedregistrering, som populært sagt går ud på at definere et warp mellem tilsvarende billeder, er forskellige antagelser om disse warpfelter, specielt at de er diffeomorphier eller styret af et linear elastisk potential, diskuteret. En ny repræsentation, som tillader statistisk analyse på et euklidisk parameterrum hvor alle afledte er defineret, er præsenteret. Det er vist at L_2-normen i parameterrummet udgør en fornuftig metrik i rummet af diffeomorphier. En ny parametrisering a 3D warpfelter til den medicinske billedregistrering, der anvender potential funktioner og Helmholtz's dekomposition i gradient og rotation er også præsenteret. Det er vist at denne repræsentation kan anses for at være en naturlig parametrisering inden for både elastisk og fluid billedregistrering, da parametrene bliver dekoplede. Gradient feltet er en første-ordens approximation til den morfologiske parameter, som determinanten af Jacobianten er.

Spline approksimationer af funktioner, og specielt af warpfelter i billedregistrering er analyseret og diskuteret. Det er vist hvordan spline baser kan læres fra optimeringsprocessen, og hvordan den kan bidrage med en fornuftig prior, der giver god regularisering a metoden. En ny eksplicit formel, baseret på den multivariate divided difference, til beregning af simplex splines er introduceret. Denne formulering tillader nem beregning af afledte, både spatielle og med hensyn til knudepositionerne. Det er vist at der

kan sættes betingelser for knudepositionerne, således at de multivariate splines stadig reproducerer polynomier, når knuderne flyttes, også selv om trianguleringen ikke er Delaunay længere. Der defineres en metode til subdivision, som ikke kræver at trianguleringerne genberegnes. Det vises hvordan disse splines kan anvendes til billedregistrering, og der indføres mål for om en forfinelse af basis funktionerne vil bidrage med bedre nøjagtighed i bestemmelse af billedwarpet.

Preface

This thesis was prepared at the Image Analysis and Computer Graphics group at DTU Informatics and submitted to the Technical University of Denmark (DTU), in partial fulfilment of the requirements for the degree of Doctor of Philosophy, Ph.D., in Applied Mathematics.

The project was funded by DTU. The work herein represents selected parts of the research work carried out in the Ph.D. period. The thesis consists of an introductory and five research papers. The reader is expected to have skills in linear algebra, mathematics, statistics and optimization on a basic university level to get a complete understanding of the covered material.

Part of the research was conducted at Technischen Universität München, under the supervision of Professor Nassir Navab, and a part was conducted on Imperial College London, under the supervision of Daniel Rueckert. The project was supervised by Professor Rasmus Larsen, DTU Informatics, and Professor Bjarne Ersbøll, DTU Informatics.

Lyngby, August 2009

Michael Sass Hansen

Publications

A number of papers have been completed during the thesis work, and 5 of them are included in the three parts of the thesis.

- *Part I: Support vectors for novelty detection and clustering*
  - Hansen, Michael Sass ; Sjöstrand, Karl; Larsen, Rasmus *On the Regularization Path of the Support Vector Domain Description.* Pattern Recognition Letters (submitted) , 2009.
  - Hansen, Michael Sass; Holm, David Alberg; Sjöstrand, Karl; Ley, Karsten Dan; Rowland, Ian John; Larsen, Rasmus *Multiscale Hierarchical Support Vector Clustering.* International Symposium on Medical Imaging 2008
- *Part II: Parametrization of deformations*
  - Hansen,Michael Sass; Hansen,Mads Fogtman; Larsen, Rasmus. *Diffeomorphic Statistical Deformation Models.* Proceedings of IEEE 11th International Conference on Computer Vision (p. 2626-2633) Presented at: Workshop on Non-rigid Registration and Tracking through learning, 2007
  - Hansen, Michael Sass; Christensen, Niels Vorgaard; Larsen, Rasmus *Curl-gradient Image Warping: Introducing Deformation Potentials for Medical Image registration using Helmholtz Decomposition.* International Conference on Computer Vision Theory and Applications 2009
- *Part III: Multivariate splines*

– Hansen, Michael Sass; Glocker, Benjamin; Navab, Nassir; Larsen, Rasmus. *Adaptive Parametrization of Multivariate B-splines for Image Registration.* IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2008

A complete list of produced papers follows.

Journal papers

[49] Hansen, Michael Sass; Sjöstrand, Karl; Larsen, Rasmus On the Regularization Path of the Support Vector Domain Description. Pattern Recognition Letters (submitted) , 2009.

[95] Sjöstrand, Karl; Hansen, Michael Sass; Larsson, Henrik B. W.; Larsen, Rasmus A path algorithm for the support vector domain description and its application to medical imaging. Medical Image Analysis, 11(5), 417-428, 2007.

Conference papers

[47] Hansen, Michael Sass; Mosbech, Thomas Hammershaimb; Ersbøll, Bjarne Kjær; Larsen, Rasmus Estimating Local Warp Field Confidence. International Symposium on Medical Imaging 2010 (submitted)

[64] Ólafsdóttir, Hildur; Pedersen, Henrik; Hansen, Michael Sass; Lyksborg, Mark; Darkner, Sune; Larsen, Rasmus Registration-based interpolation applied to cardiac MRI. International Symposium on Medical Imaging 2010 (submitted)

[116] Zikic, Darko; Glocker, Benjamin; Hansen, Michael Sass; Kameme, Ali; Navab, Nassir. Construction of Statistical Shape Models from Minimal Deformations IJ - 2008 MICCAI Workshop - Manifolds in Medical Imaging: Metrics, Learning and Beyond

[42] Hansen, Michael Sass; Christensen, Niels Vorgaard; Larsen, Rasmus Curl-gradient Image Warping: Introducing Deformation Potentials for Medical Image registration using Helmholtz Decomposition. International Conference on Computer Vision Theory and Applications 2009

[46] Hansen, Michael Sass; Holm, David Alberg; Sjöstrand, Karl; Ley, Carsten Dan; Rowland, Ian John; Larsen, Rasmus Multiscale Hierarchical Support Vector Clustering. International Symposium on Medical Imaging 2008

[41] Hansen, Mads Fogtmann; Hansen, Michael Sass; Larsen, Rasmus Conditional Statistical Model Building. International Symposium on Medical Imaging 2008

[65] Larsen, Rasmus; Sjöstrand, Karl; Hansen, Michael Sass; Ersbøll, Bjarne Kjær. KEYNOTE ADDRESS: Predicting clinical outcome from image derived morphological change. Proceedings of 2nd the International Workshop on Image Analysis and In-Vivo Pharmacology 2008

[109] Wachinger, Christian; Glocker, Benjamin; Zeltner, Jochen; Paragios, Nikos; Komodakis, Nikos; Hansen, Michael Sass; Navab, Nasir. Deformable Mosaicing for Whole-Body MRI. International Conference on Medical Image Computing and Computer Assisted Intervention MICCAI 2008

[117] Zikic, Darko; Hansen, Michael Sass; Glocker, Benjamin; Kameme, Ali; Larsen, Rasmus; Navab, Nassir. Computing Minimal Deformations: Application to Construction of Statistical Shape Models. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2008

[43] Hansen, Michael Sass; Glocker, Benjamin; Navab, Nassir; Larsen, Rasmus. Adaptive Parametrization of Multivariate B-splines for Image Registration. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) 2008

[40] Hansen, Michael Sass; Sjöstrand, Karl; Olafsdóttir, Hildur; Larsson, Henrik B. W.; Stegmann, Mikkel Bille; Larsen, Rasmus; Ersbøll, Bjarne Kjær; Heikkilä, Janne; Austvoll, Ivar; Nyström, Ingela. Robust Pseudo-Hierarchical Support Vector Clustering. Proceedings of Scandinavian Conference on Image Analysis 2007

[63] Ólafsdóttir, Hildur; Hansen, Michael Sass; Sjöstrand, Karl; Darvann, Tron Andre; Hermann, Nuno V.; Oubel, Estanislao; Ersbøll, Bjarne Kjær; Larsen, Rasmus; Frangi, Alejandro F.; Larsen, Per; Perlynn, Chad A.; Morriss-Kay, Gillian M.; Kreiborg, Sven; Kjær Ersbøll, Bjarne; Heikkilä, Janne; Austvoll, Ivar; Nyström, Ingela. Sparse Statistical Deformation Model for the Analysis of Craniofacial Malformations in the Crouzon Mouse. Proceedings of Scandinavian Conference on Image Analysis 2007

[48] Hansen, Michael Sass; Olafsdóttir, Hildur; Darvann, Tron Andre; Hermann, Nuno V.; Oubel, Estanio; Larsen, Rasmus; Ersbøll, Bjarne Kjær; Frangi, Alejandro F.; Larsen, Per; Perlyn, Chad A.; Morris-Kay, Gillian M.; Kreiborg, Sven; Wernick, Miles; Fessler, Jeffrey A.. Estimation of independent non-linear deformation modes for analysis of craniofacial malformations in crouzon mice. IEEE International Symposium on Biomedical Imaging 2007

[45] Hansen, Michael Sass; Hansen, Mads Fogtman; Larsen, Rasmus. Diffeomorphic Statistical Deformation Models. IEEE 11th International Conference on Computer Vision (p. 2626-2633) Presented at: Workshop on Non-rigid Registration and Tracking through learning, 2007

[51] Hansen, Michael Sass; Zhao, Fei; Zhang, Honghai; Walker, Nicholas E.; Wahle, Andreas; Scholz, Thomas; Sonka, Milan. Detection of Connective Tissue Disorders from 3D Aortic MR Images Using Independent Component Analysis. 2nd international workshop on Computer Vision Approaches to Medical Image Analysis, CVAMIA'06

[50] Hansen, Michael Sass; Zhao, Fei; Zhang, Honghai; Ersbøll, Bjarne Kjær; Wahle, Andreas; Scholz, Thomas; Sonka, Milan. Diagnosis of Connective Tissue Disorders based on Independent Component Analysis of Aortic Shape and Motion from 4D MR Images. The 1st International Workshop on Computer Vision for Intravascular and Intracardiac Imaging. CVII 2006

[39] Hansen, Michael Sass; Ólafsdóttir, Hildur; Sjöstrand, Karl; Larsson, Henrik B.; Stegmann, Mikkel Bille; Larsen, Rasmus. Ischemic Segment Detection using the Support Vector Domain Description. International Symposium on Medical Imaging 2007

Acknowledgements

I wish to thank my supervisors Rasmus Larsen and Bjarne K. Ersbøll for continuous guidance, both during the times of excitement as well as the times of frustration. I am much indebted by the support I received for all my different endeavors, be it academic or travel.

My thanks to the all colleagues from Technical University of Denmark, who have helped to make these years of hard labor enjoyable and interesting.

I specially want to thank my collaborators at DTU Informatics, who have been giving me many hours of interesting discussions, and many useful comments. Also I want to thank Tron A. Darvann, Nuno V. Hermann and Sven Kreiborg from the 3Dlab for providing data, showing interest in my work and showing enthusiasm, which all helped me a lot to get a good start of my thesis work.

A special thanks to Daniel Rueckert for facilitating my visit to Visual Information Processing group at Imperial College London and for many interesting discussions during my stay. I also want to thank the people at his lab for collaboration, and for giving me a good time in London.

I am also grateful to Nassir Navab for letting me visit the Chair for Computer Aided Medical Procedures & Augmented Reality at Technische Universität München, and for providing a truly inspiring environment during my stay. I wish to thank the people from his lab for the collaboration on numerous interesting projects, and for all the good times we had in München and elsewhere.

I also want to thank my remaining friends, who put up with me during my thesis work,

and to my family for their endless love and support.

Contents

CHAPTER 1

Overview

The OECD predicts that public health care costs in the member countries will increase from an average of 7 % of the gross domestic products in 2005 to exceed 12 % in 2050. This overgrowth of expenses is partly due to longer life expectancies and survival rates after critical diseases. However, the overgrowth will mainly be caused by implementation of new advanced technology for diagnostic support, monitoring, and treatment. This fact both creates a need for financing health care costs and a huge market for health care technology to be exploited. In order to ensure access to and financing of continued high quality health care in Denmark, emphasis should be put on research, development, and commercialization of health care technology in the world market. The work presented here seek to face several challenges in the analysis of medical image analysis. More precisely it consists of contributions in three areas.

In machine learning the term *unsupervised learning* is used to describe the class of problems in which one attempts to discover how unlabelled data are organized. However, in this context the *no free lunch theorem* may be phrased that you cannot make inference without making assumptions [112], and even unsupervised learning makes assumptions about the unlabelled data. In this thesis different approaches to analyzing unlabelled data are treated.

The support vector domain description (SVDD) separates unlabelled data in inliers and outliers, and may be applied for support vector clustering (SVC). Contributions to both methods are included in Part I. Despite being unsupervised, these methods are

still making assumptions about the data, and finally it is attempted to obtain a better understanding of the assumptions implicitly made about the data.

The image registration problem may be considered the unsupervised attempt to establish correspondences between images, or between an image and prior information, ie. in the form of an anatomical atlas. Unlike shape models, image registration algorithms are only making weak assumptions on the warps, which in this treatment are assumed to be described by *parameterized deformations*. Different assumptions on the warp fields, namely diffeomorphism and a linear elastic potential in the form of regularization are discussed in Part II, and attempts to develop parameterizations suited for the assumptions are presented.

In Part III spline approximations of functions, and in particular image registration warp fields are discussed. It is shown how spline bases may be learned from the optimization process, ie. image registration optimization, and how this may contribute with a reasonable prior, or regularization in the method. Spiked by the encouraging results and theory in this field, new formulae and methods are developed with the aim of increasing the utility of the treated multivariate splines.

Part I

Support vectors for clustering and outlier detection

CHAPTER 2

Introduction to support vectors

The support vector machines close to their current form were first introduced by Boser, Guyon and Vapnik in 1992 [8], and the interest has since exploded, also due to the introduction of support vector regression by Vapnik in 1995 [106]. A *dblp* search on *support vector* shows a listing of 2717 papers, and this has given rise to several views on support vectors. An interpretation that has relevance for the support vector domain description (SVDD), treated in this part of the thesis, will be put forward in the current section.

Define f by some parameterized, sufficiently smooth (this loose definition suffices for the current presentation and aim), function $f : \mathbb{R}^k \rightarrow \mathcal{H}$, where $\mathbb{R}^k$ is the k-dimensional space of parameters and $\mathcal{H}$ is the set of functions $g : \mathbb{R}^n \rightarrow \mathbb{R}$ mapping from an n-dimensional input space to the real line. Then the following optimization problem is an instructive example to examine, which could certainly be more general, however it still specializes neatly to SVDD, for which the conclusions are also valid.

$$\min_{\beta} \sum_{i=1}^{n} \left[f(\boldsymbol{x}_i; \beta) \right]_+ + \lambda \left\| \beta \right\|_{\Sigma}^2 \ , \ \beta \in \mathbb{R}^k \ , \ \boldsymbol{x}_i \in \mathbb{R}^n \tag{2.1}$$

where λ is a regularization parameter, $\| \cdot \|_{\Sigma}^2$ denotes the inner product with respect to the matrix Σ. The hinge loss $[z]_+ = (1 - z)H(1 - z)$, where $z \in \mathbb{R}$ and H is the Heavyside step function ($H(a) = 1$ for $a \geq 0$ and $H(a) = 0$ otherwise), is a

loss function, which is always zero for $z > 1$. The hinge loss shall be examined more closely since it gives (2.1) some interesting properties. In Figure 2.1, this error function is plotted along with a quadratic loss function, which measures the quadratic distance to 1 and the negative binomial log-likelihood $\log(1 + \exp[f(\boldsymbol{x}_i; \beta)])$, which is the loss function of logistic regression [55].

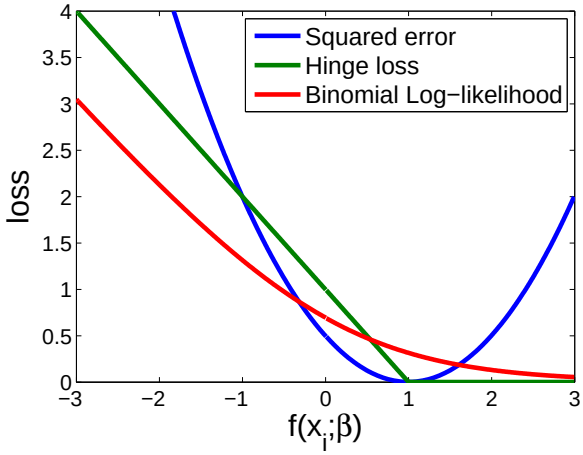


Figure 2.1: Comparison of the support vector loss function – the hinge loss, the squared error loss function and the logistic regression loss function - the binomial log-likelihood. As remarked by Hastie at al., the log-likelihood shares asymptotic behavior with the hinge loss but has a smooth transition [55].

One particular thing about the hinge loss, which is also clear from Figure 2.1, is that for function values of f larger than 1, the loss is independent of the particular value of f. Assume that f and the set of points $\{\boldsymbol{x}_i\}$ is such that a unique set of parameters $\beta^* \in \mathbb{R}^k$, different from the trivial zero vector, minimizes the problem (2.1), and furthermore ensures that all $f(\boldsymbol{x}_i; \beta^*) \geq 0$. Then β^* must be independent of the values of all $\chi = \{\boldsymbol{x}_i | f(\boldsymbol{x}_i; \beta^*) > 1, \boldsymbol{x}_i \in \{\boldsymbol{x}_i\}\}$, ie. for which $f(\boldsymbol{x}_i; \beta^*)$ is larger than one. This can be realized by seeing that the sum in (2.1) does not change with a small perturbation in $\boldsymbol{x}_i$. If f is smooth, as assumed, the loss function will still be the same value at a different $\boldsymbol{x}_i$, since the hinge loss of $\boldsymbol{x}_i$ is still 0. If a different β^n exists resulting in a smaller value of (2.1), this value would by parallel arguments also be valid for β^*, which is a contradiction with the optimality assumption, and it is therefore concluded that the parameters β^* are independent of χ. Since the zero vector was not a solution, the function value of some points $\boldsymbol{x}_{sv}$ must be 1 exactly, because this would still make $f(\boldsymbol{x}_{sv}; \beta^*) = 0$, but in this case a small perturbation of $\boldsymbol{x}_{sv}$ may still lead to a change in the function value, if it is greater than zero after the perturbation, and it cannot be concluded that β^* is independent of these points $\boldsymbol{x}_{sv}$.

In summary β^* can be shown to be independent of all points, but the *boundary* points $\{\boldsymbol{x}\} = \{\boldsymbol{x}_j | f(\boldsymbol{x}_j; \beta^*) = 1, \boldsymbol{x}_j \in \{\boldsymbol{x}_i\}\}$, which are called the *support vectors* because they are in effect the support of β^* that give β^* its value. The support vectors may still have a different impact on β^*, where impact is imagined to be measured as the effect on β^* from making a small perturbation to a support vector $\boldsymbol{x}_{sv}$. For some points the impact may be small.

Now assume that for some points $\boldsymbol{x}_{bsv}$ the function f is smaller than 1, ie. $f(\boldsymbol{x}_{bsv}; \beta^*) < 1$, following the arguments from the previous paragraph, it is seen that the points $\boldsymbol{x}_{bsv}$ all have an impact on β^*. For the special SVDD cases of f, with quadratic functions, the impact of these points is indeed equal to the maximum impact that a support vector may have, and they are therefore termed *bounded support vectors*. The interior points or *inliers*, the support vectors and the bounded support vectors are illustrated in Figure 2.2.

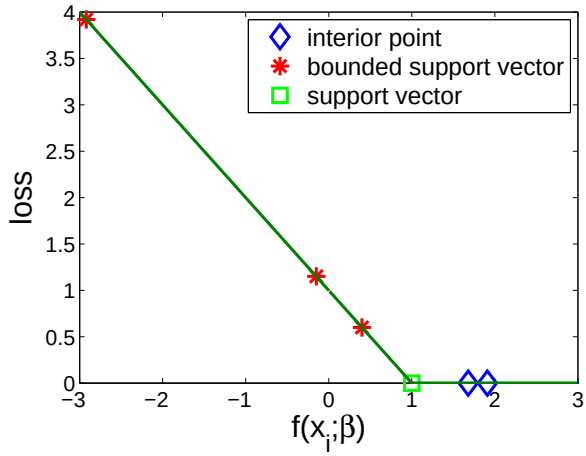


Figure 2.2: Illustration of the three classes of points, measured by their impact on the function (2.1).

In SVDD (this is, in fact, also valid for support vector machines (SVMs)) the hinge loss can be replaced by a set of slack variables and complimentary constraints on these variables. As demonstrated in Chapter 3, these can be removed by introducing Lagrange multipliers and solving the dual problem. These Lagrange multipliers are effectively determining the *impact* of the parameters, and they behave as described. An effect of this is that the parameters are independent of all the inliers, and if the number of support vectors and bounded support vectors is low, this gives a very sparse representation of the parameters.

In conclusion the hinge loss is a tool for modelling outliers and boundary, as noted by Tax and Duin [3]. These conclusions also hold for support vector regression [96]. Pontil states that the information needed for classification lies in the support vectors [81].

In the paper in Chapter 3 [49], the effect of the regularization parameter λ on the parameters is shown to be linear. The chapter introduces an efficient algorithm for calculating the entire regularization path, with the same complexity as the algorithm for the SVM by Hastie et al. [54]. In Chapter 4 an application of the entire regularization path for hierarchical support vector clustering is demonstrated [46]. In chapter 5 Bayesian methods are applied to SVDD, hinting that more informed estimation of the parameters for the support vector clustering could be obtained. Finally, in Chapter 6 the SVDD method and the obtained results are discussed.

CHAPTER 3

On the Regularization Path of the Support Vector Domain Description

Michael Sass Hansen, Karl Sjöstrand and Rasmus Larsen [1]

Abstract

Through the internet and a growing number of increasingly sophisticated measuring devices, there is a vast amount of data in many applications. However, the dimensionality is often high, and the time available for manual labelling is often scarce. Methods for unsupervised novelty detection are a great step towards meeting these challenges, and the support vector domain description has already shown its worth in this field. The method has recently received more attention, since it has been shown that the regularization path is piece-wise linear, and can be calculated efficiently. The presented work restates the new findings in a manner which permits the calculation with $O(n \cdot n_B)$ complexity in each iteration step instead of $O(n^2 + n_B^3)$, where n is the number of data points and n_B is the number of boundary points. We believe this will further promote the use of this method.

[1] This paper was submitted to Pattern Recognition Letters [49]

3.1 Introduction

We are often faced with data of high dimensionality. Imaging devices with an intrinsic high number of variables are emerging for more and more applications, and in order to deal with this class of data, a whole series of data analysis tools have emerged. Many of these use the kernel trick to create efficient algorithms dealing seamlessly with the high number of dimensions through inner products, while keeping flexibility for modelling distributions [106]. The support vector domain description (SVDD), introduced by Tax and Duin [101] in 1999, is a method for one-class labelling, which also falls into the aforementioned category. SVDD may be used for novelty detection, clustering or outlier detection [115, 5, 36]. The data is classified as either inliers or outliers through the introduction of a minimal containing sphere. The description has strong ties to the one-class version of the two-class method support vector machines (SVM) [88].

The basic goal of SVDD is to find a minimal sphere containing inliers while minimizing the distance from the boundary to the outliers. More formally it can be stated as the following optimization problem

$$\min_{R^2,\, \mathbf{a},\, \xi_i} \sum_i \xi_i + \lambda R^2 \text{ where } (\mathbf{x}_i - \mathbf{a})(\mathbf{x}_i - \mathbf{a})^T \leq R^2 + \xi_i \ , \ \xi_i \geq 0 \ \forall \ i \quad (3.1)$$

where $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_n]$ is the data matrix with each point $\mathbf{x}_i \in R^p$, $\mathbf{a}$ is the center and R is the radius of the sphere, and ξ_i are the slack variables, allowing some points, the *outliers*, to lie outside the sphere, while still satisfying the constraints. The regularization is governed by the parameter λ. A large value of λ puts a high penalty on the radius and results in a small sphere, whereas a small λ lets the radius grow to include more points as *inliers*.

Originally, the optimization problem as posed in section 3.2, is transformed into the dual problem using the Lagrange multipliers with the Karush-Kuhn-Tucker conditions, and is solved as a quadratic optimization problem. Recently it was shown by Sjöstrand et al. [95] that the regularization path of the parameter λ is piece-wise linear, and can be calculated with an $O(n_B^3 + n^2)$ complexity for each iteration step, where $n_B << n$ is the number of points on the boundary of the sphere and n is the total number of points. This result has been used to construct a generalized distance by Hansen et al. [40]. In section 3.3 a more efficient approach reducing the complexity to $O(n \cdot n_B)$ in each iteration step is derived.

3.2 The support vector domain description

A Lagrangian operator can be used to solve the problem of finding the optimum sphere, posed in (3.1). The Lagrangian is given by

$$L_p : \sum_i \alpha_i(\mathbf{x}_i\mathbf{x}_i^T - 2\mathbf{a}\mathbf{x}^T + \mathbf{a}\mathbf{a}^T - R^2 - \xi_i) + \sum_i \xi_i + \lambda R^2 - \sum_i \gamma_i\xi_i, \quad (3.2)$$

where α_i and γ_i are the Lagrange multipliers. The Karush-Kuhn-Tucker complimentary conditions hold since the optimization problem is convex, and they are given by

$$\alpha_i(\mathbf{x}_i\mathbf{x}_i^T - 2\mathbf{a}\mathbf{x}_i^T + \mathbf{a}\mathbf{a}^T - R^2 - \xi_i) = 0 \quad (3.3)$$

$$\gamma_i\xi_i = 0\,. \quad (3.4)$$

The optimum is given where the derivatives of the variables are zero

$$\frac{\delta L_p}{\delta R^2} = 0 \quad \Leftrightarrow \quad \lambda = \sum_i \alpha_i \quad (3.5)$$

$$\frac{\delta L_P}{\delta \mathbf{a}} = 0 \quad \Leftrightarrow \quad \mathbf{a} = \frac{\sum_i \alpha_i \mathbf{x}_i}{\sum_i \alpha_i} \quad (3.6)$$

$$\frac{\delta L_P}{\delta \xi_i} = 0 \quad \Leftrightarrow \quad \lambda_i = 1 - \alpha_i \quad (3.7)$$

From equations (3.7), (3.3) and (3.4), it is seen that $\alpha_i = 1$ for outliers (since $\gamma_i = 0$) and $\alpha_i = 0$ for inliers. On the boundary, α_i can take any value in $[0; 1]$. Inserting equations (3.5-3.7) in (3.2), the minimization problem is transformed to the problem of maximizing the Wolfe dual form

$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i \mathbf{x}_i\mathbf{x}_i^T - \frac{1}{\lambda}\sum_i\sum_j \alpha_i\alpha_j\mathbf{x}_i\mathbf{x}_j^T,\ 0 \le \alpha_i \le 1,\ \sum_i \alpha_i = \lambda,$$

The dimensionality of the input vectors $\mathbf{x}_i$ can be increased using a basis expansion and the dot-product substituted by an inner product. The inner products can then be replaced by $K_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j)$, where K is a positive definite kernel function satisfying Mercer's theorem. The Gaussian kernel $K_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j) = \exp -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\gamma}$ is a popular example of such a kernel function. The optimization problem may then be stated as

$$W_d = \max_{\boldsymbol{\alpha}} \sum_i \alpha_i K_{i,i} - \frac{1}{\lambda}\sum_i\sum_j \alpha_i\alpha_j K_{i,j} \quad (3.8)$$

$$0 \le \alpha_i \le 1,\ \sum_i \alpha_i = \lambda. \quad (3.9)$$

For a given λ, the squared distance from the center of the sphere to a point $\mathbf{x}$ is

$$f(\mathbf{x}; \lambda) = K(\mathbf{x}, \mathbf{x}) - \frac{2}{\lambda}\sum_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + \frac{1}{\lambda^2}\sum_i\sum_j \alpha_i\alpha_j K_{i,j}\,, \quad (3.10)$$

where the decision boundary is not necessarily a sphere in the space of the input points, although it is, in the space of the basis of the kernel function used. For the derivation the following sets are defined; the set A contains all the input points, B denotes the set of points on the boundary, O is the set of outliers, and let I be the set of inliers.

3.3 Calculating the regularization path of the SVDD

This derivation is the main contribution of the current work, and differs from the derivation by Sjöstrand et al. to provide the basis for a more efficient calculation of the parameters using updating and downdating of a matrix inverse. Two well known theorems, showing that the Lagrange multipliers are continuous for a convex problem, are stated in Appendix 3.7.1. In Section 3.3.1 an expression for the piece-wise linear relation between $\boldsymbol{\alpha}$ and λ is derived along with a scheme for fast calculation. Finally the algorithm is outlined in Section 3.3.2.

3.3.1 Piece-wise linear regularization path

Let the generalized radius be denoted by R, then a boundary point $\boldsymbol{x}_h$, where $h \in B$ must satisfy

$$f(\mathbf{x}_h;\lambda) = K_{h,h} - \frac{2}{\lambda}\sum_i \alpha_i K_{h,i} + \frac{1}{\lambda^2}\sum_i\sum_j \alpha_i\alpha_j K_{i,j} = R^2, \;\; h \in B. \quad (3.11)$$

The first sum can be split in terms depending on λ and constant terms (always 1 or 0 for points on the outside and inside). This gives $\sum_i \alpha_i K_{h,i} = \sum_{i\in B} \alpha_i K_{h,i} + \sum_{i\in O} \alpha_i K_{h,i}$. Only the first term depends on λ while the boundary set, B, stays fixed, since α_i is always 1 on the outside. Let $k_i = K_{i,i}$ and define

$$R' = R^2 - \frac{1}{\lambda^2}\sum_i\sum_j \alpha_i\alpha_j K_{i,j}$$

and notice that R' takes the same value for all $h \in B$. Let $K_{B,B}$ denote the matrix containing the inner products of the boundary points, $K_{B,O}$ denote the matrix with inner products of the boundary points and outliers, and let k_B be a vector with elements $k_i,\ i \in B$. Let $\boldsymbol{\alpha}_B$ be a vector with the Lagrange multipliers α_i on the boundary, and let $\mathbf{1}_j$ be a column vector of length j, with all elements equal to 1. Let n_B denote the number of points in B, then the set of equations (3.11) can be rewritten in matrix form as

$$\left[\frac{2}{\lambda}K_{B,B}\ \mathbf{1}_{n_B}\right]\left[\begin{array}{c}\alpha_B\\ R'\end{array}\right] = k_B - \frac{2}{\lambda}K_{B,O}\mathbf{1}_{n_O}. \quad (3.12)$$

This system of equations consists of n_B equations and n_B+1 unknown variables. The constraint from (3.5) is included in the linear system, and $\sum_i \alpha_i = \sum_{i\in B} \alpha_i + n_O$, where n_O is the number of outliers.

$$\begin{aligned}\begin{bmatrix} \frac{2}{\lambda}K_{B,B} & \mathbf{1}_{n_B} \\ \mathbf{1}_{n_B}^T & 0 \end{bmatrix} \begin{bmatrix} \alpha_B \\ R' \end{bmatrix} &= \begin{bmatrix} k_B - \frac{2}{\lambda}K_{B,O}\mathbf{1}_{n_O} \\ \lambda - n_O \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\lambda}I_{n_B\times n_B} & \mathbf{0}_{n_B} \\ \mathbf{0}_{n_B}^T & 1 \end{bmatrix} \left(\lambda \begin{bmatrix} k_B \\ 1 \end{bmatrix} + \begin{bmatrix} -2K_{B,O}\mathbf{1}_{n_O} \\ -n_O \end{bmatrix} \right)\end{aligned}$$

This may be rewritten

$$\begin{bmatrix} 2K_{B,B} & \mathbf{1}_{n_B} \\ \mathbf{1}_{n_B}^T & 0 \end{bmatrix} \begin{bmatrix} I_{n_B\times n_B} & \mathbf{0}_{n_B} \\ \mathbf{0}_{n_B}^T & \lambda \end{bmatrix} \begin{bmatrix} \alpha_B \\ R' \end{bmatrix} = \lambda \begin{bmatrix} k_B \\ 1 \end{bmatrix} + \begin{bmatrix} -2K_{B,O}\mathbf{1}_{n_O} \\ -n_O \end{bmatrix}$$

Define

$$K' = \begin{bmatrix} 2K_{B,B} & \mathbf{1}_{n_B} \\ \mathbf{1}_{n_B}^T & 0 \end{bmatrix}$$

Assuming the points are in general position in the expanded basis, such that the circle center is determined by at most the expanded plus one points, then K' can be inverted to obtain an expression for $\boldsymbol{\alpha}_B$.

$$\begin{bmatrix} I_{n_B\times n_B} & \mathbf{0}_{n_B} \\ \mathbf{0}_{n_B}^T & \lambda \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_B \\ R' \end{bmatrix} = K'^{-1} \left(\lambda \begin{bmatrix} k_B \\ 1 \end{bmatrix} + \begin{bmatrix} -2K_{B,O}\mathbf{1}_{n_O} \\ -n_O \end{bmatrix} \right) \quad (3.13)$$

From this we learn that $\boldsymbol{\alpha}_B$ is piece-wise linear in λ, while none of the constraints given in (3.9) are violated.

3.3.2 The algorithm

Since α_i, by theorem 3.2, is continuous as a function of λ, this may be applied in finding the regularization path. Notice that if $\lambda = n$, it is easily seen that $\alpha_i = 1\ ,\ i = 1, \ldots, n$. Therefore the algorithm is started in a state, where $\lambda = n$, and from this starting point λ can be decreased, and the two events that happen while decreasing λ are

- A point from either the inside or the outside enters the boundary
- A point exits the boundary to either the inside or the outside.

In between any of these events, the regularization path is piece-wise linear, as shown in section 3.3.1, and the parameters can be calculated from (3.13).

In the following, let l be the last event that occurred and $l+1$ be the next event, so that λ^l was the previous and bigger value of the regularization parameter. Let $\boldsymbol{\alpha}_l$ be the

value of all α_i at the event l and $\boldsymbol{\alpha_{l+1}}$ at the following event $l+1$. Then using that $\boldsymbol{\alpha}$ is continuous

$$\boldsymbol{\alpha}_{l+1} = \boldsymbol{\alpha}_l + (\lambda^{l+1} - \lambda^l)\boldsymbol{p}_l, \tag{3.14}$$

where only the points α_i on the boundary need to be updated. Let $\boldsymbol{x}_e \in A$ be any point, and λ_e be the value of λ for which the event following event l would happen, if everything except λ was fixed. In Section 3.3.2.1 λ_e is found for all points outside the boundary, ie. $I \cup O$, and in Section 3.3.2.2 λ_e is found for points on the boundary B.

3.3.2.1 Boundary entry event

This event happens at a point where the distance to one of the non-boundary points equals the radius of the (generalized) sphere. This condition can be formulated as

$$f(\mathbf{x}_e;\lambda) - R^2 = K_{e,e} - \frac{2}{\lambda_e}\sum_i \alpha_i K_{e,i} + \frac{1}{\lambda_e^2}\sum_i\sum_j \alpha_i\alpha_j K_{i,j} - R^2 = 0$$

Using that R^2 is given by equation (3.11) we find that

$$\begin{aligned} 0 &= K_{e,e} - \frac{2}{\lambda_e}\sum_i \alpha_i K_{e,i} - K_{h,h} + \frac{2}{\lambda_e}\sum_i \alpha_i K_{h,i} \\ &= K_{e,e} - K_{h,h} + \frac{2}{\lambda_e}(K_{h,A} - K_{e,A})(\boldsymbol{\alpha}_l + (\lambda_e - \lambda^l)\boldsymbol{p}_l) \\ \Leftrightarrow \lambda_e - \lambda^l &= -\frac{(K_{h,A} - K_{e,A})\boldsymbol{\alpha}_l + \frac{\lambda^l}{2}(K_{e,o} - K_{h,h})}{(K_{h,A} - K_{e,A})\boldsymbol{p}_l + \frac{1}{2}(K_{e,e} - K_{h,h})}\,, \end{aligned} \tag{3.15}$$

where the sums have been replaced by matrix products, and $\boldsymbol{\alpha}$ has been substituted using (3.14). As we are decreasing the value of λ, we are only interested in values of $\lambda_e - \lambda^l$ smaller than 0. The biggest value, smaller than zero, of $\lambda_e - \lambda^l$ therefore marks the first entry event to occur. Since the complexity of calculating $K_{A,A}\boldsymbol{\alpha}_l$ is $O(n^2)$, this calculation should be done iteratively, updating $K_{A,A}\boldsymbol{\alpha}_l$ in each step, by noting $K_{A,A}\boldsymbol{\alpha}_{l+1} = K_{A,A}\boldsymbol{\alpha}_l + (\lambda^{l+1} - \lambda^l)K_{A,B}\boldsymbol{p}$, it can be calculated with complexity $O(n \cdot n_B)$.

3.3.2.2 Boundary exit event

Though equation (3.13) gives an explicit expression for α_i, this is only the case, when i denotes a point on the boundary. Otherwise α_i is limited by the constraints $0 \leq \alpha \leq 1$. As α_i, for i on the boundary, increases or decreases monotonically, only one of the two

constraints comes into effect. Let the effective constraint be given by

$$C_{exit,e} = \begin{cases} 0 & \text{if} \quad p_e \geq 0 \\ 1 & \text{if} \quad p_e < 0 \end{cases}$$

then the boundary exit value for the eth point is given by

$$\lambda_e - \lambda^l = \frac{C_{exit,e} - \alpha_{l,e}}{p_e}$$

3.3.2.3 Finding the next event $l+1$

Having calculated the first entry event and the first exit event, the only thing left is to choose which of the two events happens first and let

$$\lambda^{l+1} = \lambda^l + \max_{\boldsymbol{x}_e \in A, \lambda_e - \lambda^l < 0} \{\lambda_e - \lambda^l\}$$

An issue that has to be dealt with is how to propagate $\boldsymbol{\alpha}$ if the boundary set is the empty set. This is done simply by adding the closest outlier to the boundary set, which corresponds to making a discontinuous change in R^2, but not in $f(\boldsymbol{x})$ or $\boldsymbol{\alpha}$.

3.4 Complexity

The slope of $\boldsymbol{\alpha}$ with respect to λ, given by $\boldsymbol{p} = K'^{-1}[k_B\ 1]^T$ in (3.13) can be calculated using simple matrix multiplications of complexity $O(n_B^2)$. K'^{-1} can be calculated using updating and downdating, also with complexity $O(n_B^2)$, as is shown in Appendix 3.7.2. The complexity of calculating $\boldsymbol{p}_B$ is $O(n_B^2)$, while the complexity of evaluating the boundary entry conditions is $O(n \cdot n_B)$, which means that the overall complexity in each iteration step is of the order of $O(n \cdot n_B)$, as $n \geq n_B$. The regularization path of the SVM could be found with the same complexity [54], and the problems also show strong resemblance. Figure 3.1 shows a graph of the calculation time of the previous algorithm and the presented implementation. Note that the computation time follows the theoretical complexity. For a population of 1000 points, the current implementation can be up to 100-times faster, and for our testing purposes this has been the limit for the length of the calculations we set up for the previous implementation. The stability of the calculations has also set a natural limit, as discussed in section 3.4.1.

3.4.1 Stability

As the current fast implementation depends strongly on updating calculations, rather than recalculating them, there is a risk that the result will drift due to numeric instability. This issue is investigated by running the implementation on different data sets, while testing the results for given values of the regularization parameter λ using an implementation of quadratic programming. The stability was tested on randomly generated data sets of dimension 2 and 3, and size $n = 3000$ and the result can be seen in Figure 3.2. The result can be seen to differ by no more than 0.5% even for $25,000$

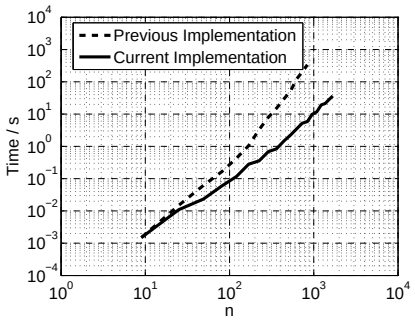


Figure 3.1: Logarithmic plot of the complexities of the two different implementations.

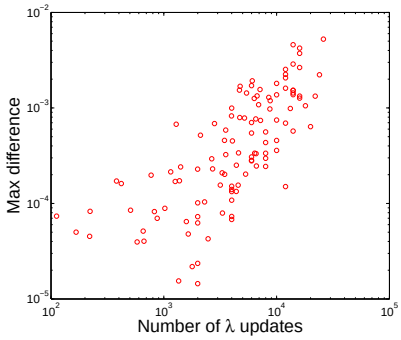


Figure 3.2: Logarithmic plot of the error of the implementation compared to the result obtained using quadratic programming.

updates and downdates of the inverse and the value of λ.

In the high n domain, the previously presented implementation suffered from imprecision in calculating the next value λ^{l+1} because the value was directly estimated rather than, as in the current work, the difference $\lambda^{l+1} - \lambda^{l}$.

3.5 Demonstration

To demonstrate the method a small example is analyzed using the implemented algorithm. From two sources with 2-dimensional Gaussian distributions 100 points are sampled and they are analyzed with a Gaussian kernel function with a width of 1. The result can be seen in Figure 3.3. Note that this value of the kernel parameter leaves room for a rather flexible decision boundary. In the Figure it can be seen that some of the points, the support vectors, are outside and some are inside, corresponding to a α_i

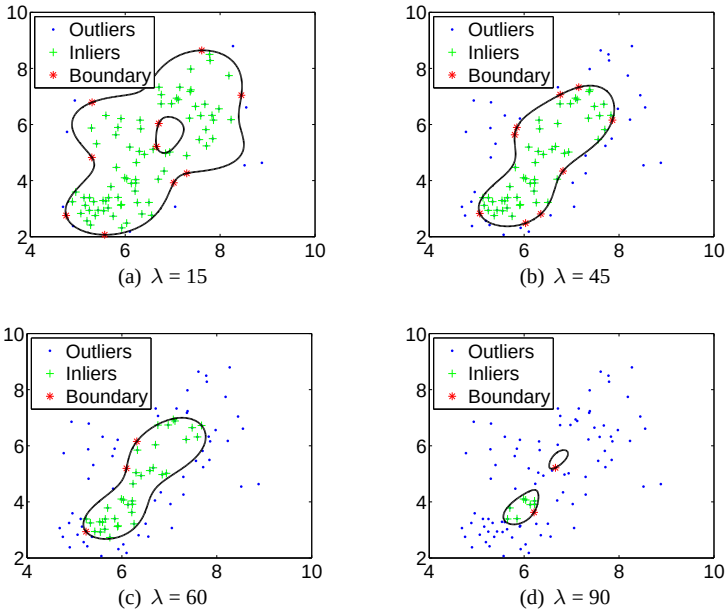


Figure 3.3: Decision boundaries for different values of λ.

of 0. In Figure 3.4 the entire regularization path of $\boldsymbol{\alpha}$, that is the α_i corresponding to each point, can be observed.

The calculation is performed in a fraction of a second for this rather small sample size.

3.6 Conclusion

The support vector domain description (SVDD) is a new and popular method. Recent work by Sjöstrand et al. [95] demonstrated that the regularization path of the weight coefficients depends piece-wise linearly on λ. This allows for an efficient calculation of the regularization path. The current work restates new findings in a manner that permits the calculation with a complexity of $O(n \cdot n_B)$ instead of $O(n^2 + n_B^3)$ in each iteration step. It has been demonstrated that for $n = 800$ points, the calculation of the regularization path could be performed up to 100-times faster. The algorithm keeps the numeric error small for sample sizes up to 3000 points, smaller than 0.5% in the analyzed cases. We believe that this contribution will allow for even more applications of the method, either for choosing robust estimates of the distance, or possibly in the

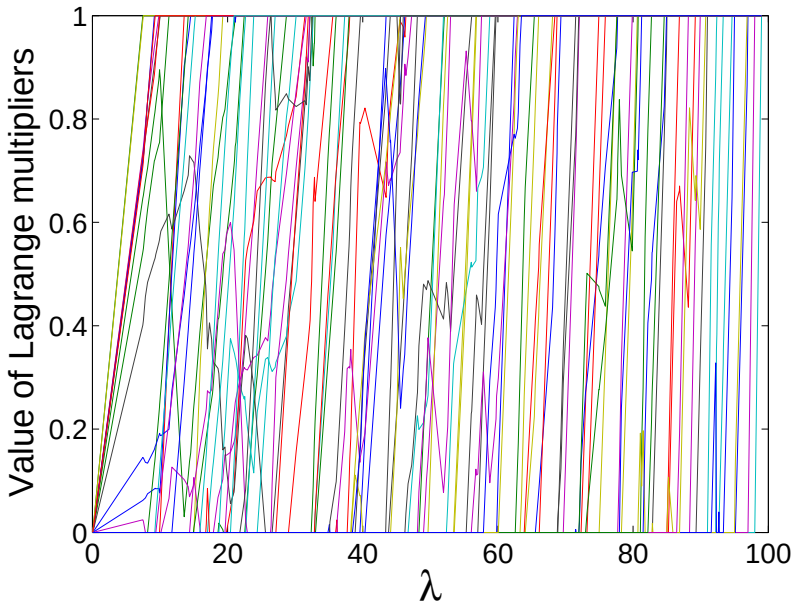


Figure 3.4: The path of the loading coefficients α_i, all going from 0 on the inside to 1 on the outside, as λ is increased.

area of support vector clustering.

3.7 APPENDIX

3.7.1 Continuity of the Lagrange multipliers

THEOREM 3.1 *The Wolfe dual form W_d given by (3.8) is continuous with respect to the regularization parameter, λ.*

PROOF. Let $\boldsymbol{\alpha}_1$ be a solution for a given set of points and regularization parameter λ_1, and $\boldsymbol{\alpha}_2$ a solution for regularization parameter λ_2. It is seen that for any $0 \leq s \leq 1$, $\boldsymbol{\alpha} = s\boldsymbol{\alpha}_1 + (1-s)\boldsymbol{\alpha}_2$, satisfies the conditions on $\boldsymbol{\alpha}$, and due to the polynomial form of W_d it can be concluded that W_d is continuous.

THEOREM 3.2 *The Lagrange multipliers $\boldsymbol{\alpha}$ are continuous with respect to λ.*

PROOF. Follows directly from the fact that W_d is continuous and the solution to a convex problem is unique.

3.7.2 Calculation of K'^{-1}

The two events that may occur to the boundary set either reduce or augment B by one point. This allows for an efficient calculation of K'^{-1}, which is the purpose of the current section. Using the following result by Volker Strassen [99], the updates and downdates of the inverse can be calculated efficiently

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + A^{-1}BS_ACA^{-1} & -A^{-1}BS_A \\ -S_ACA^{-1} & S_A \end{bmatrix} \tag{3.16}$$

where the Schur complement of A is denoted $S_A = (D - CA^{-1}B)^{-1}$. The efficient calculation of K'^{-1}_{l+1}, the inverse of matrix K' after event l, will be presented in the following two paragraphs.

Updating Suppose that the point b^* has been added to B_l to form B_{l+1}, then K'_{l+1} can be written as

$$K'_{l+1} = \begin{bmatrix} K'_l & K_{B_l,b^*} \\ K_{b^*,B_l} & K_{b^*,b^*} \end{bmatrix}. \tag{3.17}$$

Here $S_A = (K_{b^*,b^*} - K_{b^*,B_l}K'^{-1}_l K_{B_l,b^*})^{-1}$ and define $S_C = K'^{-1}_l K_{B_l,b^*}$, then the inverse can be calculated from

$$\begin{bmatrix} K'_l & K_{B_l,b^*} \\ K_{b^*,B_l} & K_{b^*,b^*} \end{bmatrix}^{-1} = \begin{bmatrix} K'^{-1}_l + S_CS_AS_CT & -S_CS_A \\ -S_AS_C^T & S_A \end{bmatrix}, \tag{3.18}$$

which only requires a multiplication of a vector with a matrix of size n_B, and this multiplication has complexity $O(n_B^2)$

Downdating Suppose that the point b^* has been removed from B_l to form B_{l+1}. Then K'^{-1}_l can be written using equation (3.18), only here b^* is the point that was removed from the boundary

$$K'^{-1}_l = \begin{bmatrix} A_{n_{B_{l+1}} \times n_{B_{l+1}}} & B_{n_{B_{l+1}} \times 1} \\ C_{1 \times n_{B_{l+1}}} & D_{1\times 1} \end{bmatrix} = \begin{bmatrix} K'_{l+1} & K_{B_{l+1},b^*} \\ K_{b^*,B_{l+1}} & K_{b^*,b^*} \end{bmatrix}^{-1}$$
$$\Rightarrow K'^{-1}_{l+1} = A - BCD^{-1} \tag{3.19}$$

CHAPTER 4

Hierarchical Multiscale Support Vector Clustering

Michael Sass Hansen, David Alberg Holm, Karl Sjöstrand, Carsten Dan Ley, Ian John Rowland and Rasmus Larsen[1]

Abstract

Clustering is the preferred choice of method in many applications, and support vector clustering (SVC) has proven efficient for clustering noisy and high-dimensional data sets. A method for multiscale support vector clustering is demonstrated, using the recently emerged method for fast calculation of the entire regularization path of the support vector domain description. The method is illustrated on artificially generated examples, and applied for detecting blood vessels from high resolution time series of magnetic resonance imaging data. The obtained results are robust while the need for parameter estimation is reduced, compared to support vector clustering.

[1]This paper was presented at the International Symposium for Medical Imaging 2007 [46]

4.1 Introduction

With the emergence of new imaging modalities and an ever-increasing resolution of existing ones, the need for unsupervised clustering is greater than ever. Clustering can be performed by parametric models, which is the case in the k-means algorithm introduced by MacQueen [70], or it can be done by using a distance measure for grouping points as in hierarchical clustering algorithms.

Ben-Hur et al. presented support vector clustering (SVC), which utilizes kernel functions to efficiently deal with high dimensional data [3]. SVC uses the one-class support vector domain description (SVDD) as the basis of the clustering algorithm. Tax and Duin introduced SVDD in 1999 [101]. In SVDD the points are mapped into a high dimensional feature space, which divides inliers from outliers. It is frequently calculated with a Gaussian kernel replacing the Euclidian inner product. The decision boundary consists of contours enclosing clusters of the data points from the surrounding outliers.

The SVC clustering algorithm makes no assumption about the number of clusters or the shape of the clusters. Ben-Hur et. al. proposed to vary the parameters of the SVDD, namely the kernel width and the regularization parameter, in a manner that increases the number of clusters while keeping the number of outliers and bounded support vectors (BSV) low. Hierarchical support vector clustering was presented by Ben-Hur in [4], and this algorithm applies SVC subsequently on each cluster of the data contained in clusters. It achieves in this way a strict hierarchy of clusters. The clustering, however, is determined from the initial clustering and its parameters.

The cluster labelling can be assigned more efficiently using proximity graph modelling as proposed by Yang et al. [114].

In line with the seminal work by Efron et al. [24], where the regularization path of the LASSO regression algorithm is derived and similar to the work on the support vector machine by Hastie et al. [54], Sjöstrand and Larsen showed in 2006 that the entire regularization path of the SVDD is piecewise linear and can be calculated efficiently [94]. This result is the backbone of a pseudo-hierarchical support vector clustering (HSVC), presented by the current authors [40]. This algorithm estimates a clustering given a scale parameter of the Gaussian kernel. Clusters are estimated robustly and efficiently for all values of the regularization parameter. A more robust clustering estimate is calculated from this ensemble of clusterings that form competing hierarchical representations, without prior knowledge of the number of outliers. The HSVC clustering was tested on registered perfusion MR images.

The current work presents multiscale support vector clustering (MSVC), which may be considered as parallel HSVC on multiple scales of the Gaussian kernel. We observe the data in different scales, and from the clustering obtained in multiple scales we propose

to learn the clustering. The proposed method is independent of the number of outliers and bases the clustering on information from all scales in the range of scales of the data.

4.2 The MSVC algorithm

The foundation of the multiscale support vector clustering algorithm is the one-class support vector classification, this is also, in general, the case for previous SVC algorithms. The recently emerged method for an efficient calculation of the entire regularization path of the SSVD forms the basis for finding cluster boundaries as described in Section 4.2.1. Between events the discrimination function varies monotonically, and it is concluded that the description is complete. The MSVC algorithm is described in Section 4.2.4.

4.2.1 Cluster Boundaries from the SVDD

The cluster boundaries defining the different clusters are based on the support vector domain description (SSVD), posing it as a quadratic optimization problem for a fixed value of the regularization parameter and given a kernel function. The criterion to be maximized, given a point set x_i, can be formulated as

$$\min_{R^2,\, \mathbf{a},\, \xi_i} \sum_i \xi_i + \lambda R^2 \quad , \qquad \text{Subject to}$$
$$(\mathbf{x}_i - \mathbf{a})(\mathbf{x}_i - \mathbf{a})^T \leq R^2 + \xi_i \quad \text{and} \quad \xi_i \geq 0 \ \ \forall\, i.$$

This criterion seeks to find the minimal sphere encapsulating most of the points, while minimizing the distance to the outliers also denoted support vectors. The regularization parameter λ helps to minimize the radius R^2 and for large values of λ the radius will tend to be smaller and vice versa. The number of outliers is strongly related to the regularization parameter λ.

Using Lagrange multipliers the optimization problem (4.1) can be restated as

$$\max_{\alpha_i} \sum_i \alpha_i \mathbf{x}_i \mathbf{x}_i^T - \frac{1}{\lambda} \sum_i \sum_j \alpha_i \alpha_j \mathbf{x}_i \mathbf{x}_j^T,$$
$$0 \leq \alpha_i \leq 1, \ \sum_i \alpha_i = \lambda, \tag{4.1}$$

where α_i are the Lagrange mulitpliers. For inliers $\alpha_i = 0$ and for outliers $\alpha_i = 1$, which is a consequence of the Karush-Kuhn-Tucker complimentary conditions. For

points on the boundary the, *support vectors*, α are between 0 and 1. The outliers are also called *bounded* support vectors. The dimensionality can be increased using a basis expansion and substituting the dot-product with an inner product, the inner products can be replaced by a suitable kernel function K, $K_{i,j} = K(\mathbf{x}_i, \mathbf{x}_j)$. In the current work the applied kernel function was the Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\gamma)$ with multiple scales. The optimization problem can be formulated similarly to the problem original stated by Tax and Duin [101], only with a slightly different notation.

$$\max_{\alpha_i} \sum_i \alpha_i K_{i,i} - \frac{1}{\lambda} \sum_o \sum_j \alpha_i \alpha_j K_{i,j}$$
$$0 \leq \alpha_i \leq 1, \sum_i \alpha_i = \lambda. \tag{4.2}$$

For a given λ the squared distance from the center of the sphere to a point $\mathbf{x}$ is

$$f(\mathbf{x}; \lambda) = \|h(\mathbf{x}) - \mathbf{a}\|^2 = K(\mathbf{x}, \mathbf{x})$$
$$-\frac{2}{\lambda} \sum_i \alpha_i K(\mathbf{x}, \mathbf{x}_i) + \frac{1}{\lambda^2} \sum_i \sum_j \alpha_i \alpha_j K_{i,j} \tag{4.3}$$

4.2.1.1 The Entire Regularization Path of the SVDD

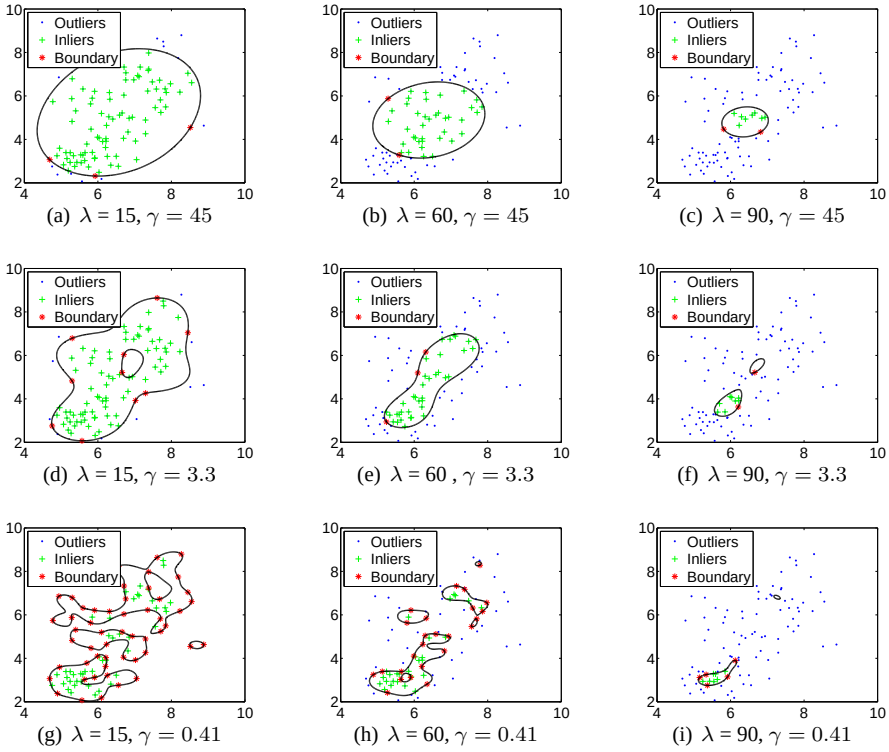
It was shown by Sjöstrand and Larsen that the entire regularization path of the parameter λ can be calculated with approximately the same complexity as required for solving the initial optimization problem, posed by Tax and Duin [94]. The regularization path of the parameters α_i is piecewise linear, which allows for the efficient calculation. This fact can be realized by analysis of the distance functions of two points on the boundary.

$$f(\mathbf{x}_h; \lambda) = f(\mathbf{x}_k; \lambda), \quad h, k \in B \tag{4.4}$$

where B is the set of points on the boundary. Formulating this equation for different points on the boundary and using the constraints on α_i from (4.2) gives a complete set of equations for estimating all α_i. Let $\boldsymbol{\alpha}$ be a vector with the values α_i and let $\boldsymbol{p}$ and $\boldsymbol{q}$ be the slope and intersection respectively, then (refer to [94] for a detailed derivation)

$$\boldsymbol{\alpha} = \lambda \boldsymbol{p} + \boldsymbol{q}, \tag{4.5}$$

where $\boldsymbol{p}$ and $\boldsymbol{q}$ are the slope and intersection of the path on the intervals between events where a point either leaves or joins the boundary, $[\lambda_l; \lambda_{l+1}[$. An example of clustering boundaries for different values of the regularization parameter and given kernel width is illustrated in Figure 4.1.



(a) $\lambda = 15, \gamma = 45$ (b) $\lambda = 60, \gamma = 45$ (c) $\lambda = 90, \gamma = 45$

(d) $\lambda = 15, \gamma = 3.3$ (e) $\lambda = 60, \gamma = 3.3$ (f) $\lambda = 90, \gamma = 3.3$

(g) $\lambda = 15, \gamma = 0.41$ (h) $\lambda = 60, \gamma = 0.41$ (i) $\lambda = 90, \gamma = 0.41$

Figure 4.1: SVDD calculated for the entire regularization path. The bold line marks the boundary between inliers and outliers, actually a sphere in the expanded basis of the kernel function.

4.2.2 Support Vector Clustering

In SVC, proposed by Ben-Hur et al. [3] the clustering is based directly on the result of the SVDD. SVC forms an initial step in our proposed multiscale support vector clustering. From the distance given by Eq. (4.3) the radius can naturally be defined as the distance of the points on the boundary. Now R of the SVDD can be calculated by

$$R = f(\mathbf{x}_k; \lambda) = K_{k,k} - \frac{2}{\lambda}\sum_i \alpha_i K_{k,i} + \frac{1}{\lambda^2}\sum_i \sum_j \alpha_i \alpha_j K_{i,j}.$$

Consider an arbitrary point $\boldsymbol{x}$, and let $g(\boldsymbol{x}, \lambda)$ be a distance function, defining the distance to the boundary.

$$g(\boldsymbol{x}, \lambda) = f(\boldsymbol{x}, \lambda) - R \ . \tag{4.6}$$

Now g is the decision function determining if a point is an inlier or an outlier. In Figure 4.1 the discriminating function g is calculated to create the contour dividing inliers from outliers. Since the optimization problem is to find a sphere in the *space of the expanded basis*, the result appears very little like a sphere in the two dimensional input space. The different enclosed areas could be considered as clusters, denoted support vector clusters.

4.2.2.1 Cluster assignments

The decision function $g(\boldsymbol{x}, \lambda)$ specifies if $\boldsymbol{x}$ is an inlier or outlier, but it contains no direct information to determine if two points are connected. Inspired from Figure 4.1 it is observed that all paths connecting two points in two different clusters have some points outside the clusters, which obviously serves as a general definition of clusters. The current algorithm uses an adjacency matrix to identify the cluster. The connection graph is sparsely built, similar to the approach chosen by Yang et. al. [114]. The adjacency matrix A is given by

$$A_{ij} = \left\{ \begin{array}{ll} 1 & , \text{ if } g(\mathbf{x}_i + \mu(\mathbf{x}_j - \mathbf{x}_i)) < 0 \ \forall \ \mu \in [0;1] \\ 0 & , \text{ else} \end{array} \right\}. \tag{4.7}$$

The clusters are detected from the adjacency matrix by using standard graph theory concepts. Outliers are by definition not adjacent to any points, but are assigned to the closest detected cluster, using a euclidian distance measure.

4.2.3 SVC based on the Entire Regularization Path

Given regularization parameter λ and kernel width γ the clustering can be determined from the adjacency matrix (4.7). When λ varies on the interval [0;n] gives rise to

changes in the distance function, and thus potentially the clustering. Previously we showed that $f(\boldsymbol{x}, \lambda)$, the distance function given by Eq. (4.3), is monotonic on the interval $[\lambda_l; \lambda_{l+1}[$ between two events [40]. From this it can be concluded that the events of a point joining or leaving the boundary of the SVDD is also important related to the formations of clusters. On the interval $[\lambda_l; \lambda_{l+1}[$ a cluster cannot be divided and merge again, meaning to say that all cluster formations will be observed by only examining the points of events.

4.2.4 Multiscale Support Vector Clustering

The width of the Gaussian kernel, γ, is effectively defining the scale of the clustering problem. When γ is of the same order as the biggest distance in the data, all point pairs produce sizeable kernel values as noted by Ben-Hur et al. [3].

$$\gamma_{max} = \max_{i,j} \|\mathbf{x}_i - \mathbf{x}_j\|^2 . \tag{4.8}$$

At this value of γ the resulting boundary will form only one cluster, which for the purpose of clustering is not very interesting, but it sets an upper limit to the scale of the clustering. As the one-cluster scenario is not very interesting we choose an initial value of the kernel 5 times smaller than γ_{max} were we expect to observe some clustering, at least for some values of the regularization parameter λ. The algorithm then continues by cutting by half the scale parameter, which provides information about all scales in the detected clusters. When most of the clusters only contain one cluster due to a small scale parameter, the algorithm is stopped.

4.2.4.1 Hierarchical Structure of Clusters

The clusters only change slowly with changes in the regularization parameter λ, which gives the different support vectors clusterings for given parameters λ and α a rather high level of redundancy. When an event consists of a point leaving the boundary to become an outlier, this does not necessarily alter the boundary much elsewhere. Since the point is still close to the same cluster, and may be associated with this, still, many clusters are close to identical. The similarity can be observed in Figure 4.1. Moreover, the same clusters may appear again at a different scale for a different value of the regularization parameters.

The idea presented in this paper, is to collect all the similar clusterings, across regularization parameter and scale parameter, and build a hierarchy of clusters. The toy example illustrated in Figure 4.1 demonstrates clustering on an increasingly smaller scale., and there is a strong relation between the different clusterings of the data, which is illustrated in Figure 4.2.

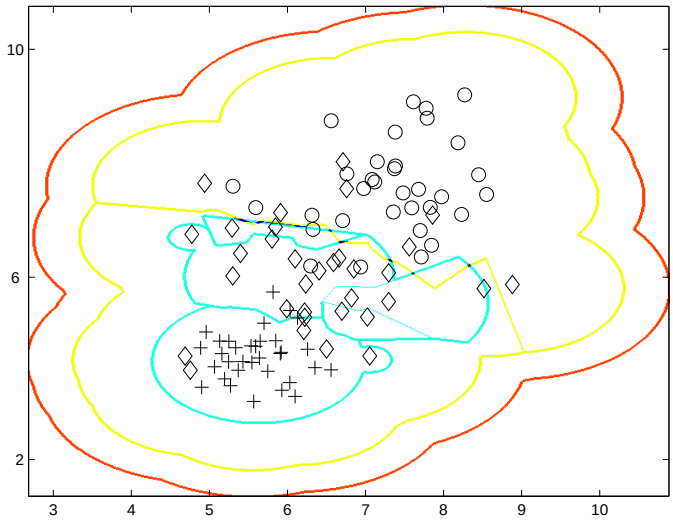


Figure 4.2: The hierarchy of multiscale support vector clustering: From coarse to detailed description.

The clusters, calculated for different values of the regularization and scale parameters, are not, in general, nested in a strictly hierarchical way. In fact multiple different hierarchical clusterings may be proposed. These different cluster representations demonstrate ways to split a cluster in smaller clusters.

4.2.4.2 Quality Measure of Competing Clusterings

The described algorithm results in several competing cluster representations of the data, and each representation has several scales. To assess the quality of different clusterings we previously introduced a scheme similar to the 'within' and the 'between' covariance matrices, trace$(S_W^{-1} S_B)$ [40]. Instead of S_W we argued that a weighted within matrix S_W^* should be calculated, weighted by the length of the interval where a given point is an inlier, or an outlier associated with the cluster.

$$S_W^* = \sum_{j=1}^{n_{\text{clusters}}} \sum_{i \in \mathbf{C}_j} \frac{1}{\Pi_j} (\mathbf{x}_i - \mu_i)^T \Lambda_{j,i} (\mathbf{x}_i - \mu_i) \,, \tag{4.9}$$

where Λ_j defines the weighting of the point, which depends linearly on the length of the interval of λ where the point is an inlier and where it is an outlier. Π_j is a nor-

malization constant. A potential clustering can now be assessed using the measure trace$({S_W^*}^{-1} S_B)$, which evaluates the variance within clusters, compared to the introduced distance between clusters. In Figure 4.1 small values of λ, corresponding to a high confidence in the data, results in a separation of the two parts of the '+' cluster, whereas the other groups are merged into one cluster. This is opposite for high values of the regularization parameter, where the smaller clusters only appear to be outliers, but the two overlapping clusters are divided. The discrimination feature removes the need to select one value of λ, and appears to adapt to clusters of different variance. The criterion for accepting a subclustering is introduced as a threshold on the cluster separation, given by trace$({S_W^*}^{-1} S_B)$. The lower the threshold, the more clusters are accepted.

4.2.5 Complexity

The efficient calculation of the entire regularization path of the SVDD allows the application of the described algorithm. The complexity for the referenced algorithm is O($n^2 + n_B^2$) for each step between two events, where n_B is the number of boundary points. For each event, the clusters are detected from the adjacency matrix, which can also be calculated with a complexity of the order of O(n^2). Comparing with other clusters is done with complexity O($n \cdot n_{clusters}$). Since the number of events is typically in the vicinity of 3-5 n the overall complexity is polynomial with a degree around 3. On the tested example, with about 100 points in 300 dimensions, the algorithm took a few minutes. The algorithm performance is virtually independent on the number of dimensions whereas it works best for a limited number of observations.

4.3 An application of MSVC: Vascular detection from MR sequences

When tumors reach a volume of $1 - 2mm^3$, the supply of nutrients through simple diffusion is not sufficient to allow further growth. The tumor response via the secretion of signalling molecules that stimulate angiogenesis (the process of developing new blood vessels). The angiogenic process is an obvious target in the design of new anticancer agents and extensive research has been conducted in the area of regulation of angiogenesis [61].

An approach to study angiogenesis in mice is placing a Matrigel chamber assay in mice in vivo [67]. To assess in vivo angiogenesis, MR images have been acquired before, during and after the injection of a contrast agent bolus, which reduces the signal of blood in order to detect blood vessels [56]. The matrigel chamber set-up is shown in

Figure 4.3, inside the mouse and after the extraction. The effects on the intravascular signal due to the bolus administration is illustrated in Figure 4.4.

(a) (b)

Figure 4.3: a: A mouse with embedded Matrigel chamber. b: A Matrigel chamber removed from a mouse. Red areas correspond to areas with blood vessels.

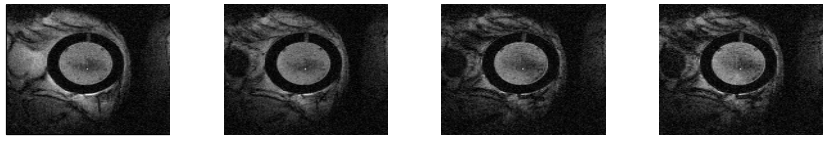


Figure 4.4: Snapshots of an MR sequence of a Matrigel chamber placed in a mouse.

300 consecutive frames are acquired with a time resolution of 2.56 seconds. Animals were anesthetized and fixated during the scan, resulting in less than 1 pixel movement throughout the scan. Due to limitations in bandwidth and memory, a subset of pixels was selected for designing the cluster criteria.

The chosen pixels, all within the matrigel chamber, are illustrated in Figure 4.5. In Figure 4.6, two time series of two different pixels are illustrated (noise pixel + pixel of interest). The difference between the noise pixel and the pixel of interest is small compared to the level of noise. Consequently MSVC was applied to the points illustrated in Figure 4.5, and three clusters were identified

Since the mouse was fixated during the scan, the frames are assumed to be registered. Because of limitations in bandwidth and memory, a subset of pixels is selected for designing the cluster criteria. The chosen pixels, all within the Matrigel chamber are illustrated in Figure 4.5. In Figure 4.6 two time series of two different pixels are illustrated. The difference between the noise pixel and the pixel of interest is small compared to the level of noise. MSVC was applied to the points illustrated in Figure 4.5. Three clusters were identified from the multiscale support vector clustering by applying the quality measure described in Section 4.2.4.2. Figure 4.7 shows average intensity curves of the points assigned to the different clusters.

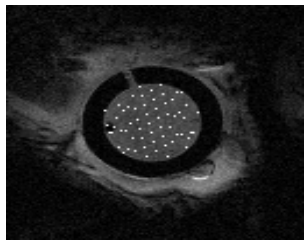


Figure 4.5: The (white) marked subset of points is used for calculation of the clustering by MSVC.

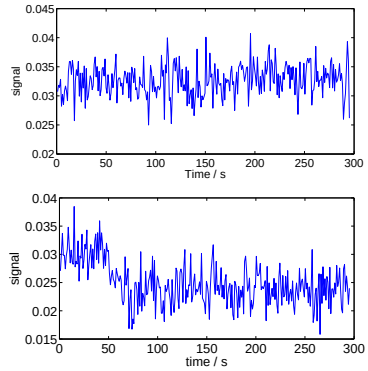


Figure 4.6: Selection of the signal response of two pixels from within the Matrigel chamber. Top: no blood vessel, only noise. Bottom: probably contains a blood vessel. The noise is of the same order as the signal change.

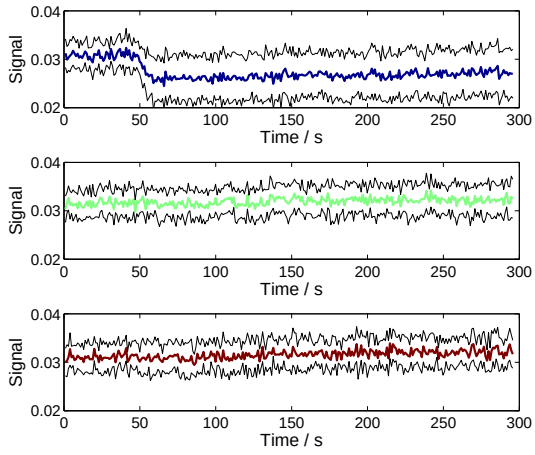
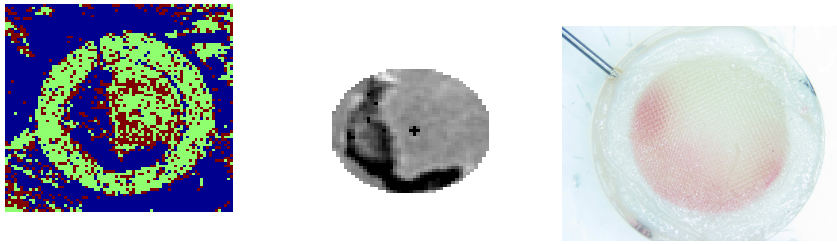


Figure 4.7: Curves showing the average of each of the three clusters $\pm$ one standard deviation. Only one appears contrast-bolus and thus blood related.

The distance to the 3 clusters is calculated pixel-wise, and points assigned to the nearest cluster. The clustered images are compared to the difference between T2 before and after injection of contrast agent (ΔT2) and to an optical image of the extracted chamber to validate the results. Figure 4.8 shows the results from the three modalities. It can be observed that for this chamber there is a very strong correlation between the blue cluster (Fig. 4.8(a)), and the ΔT2 weighted images (Fig. 4.8(b)) and the red area on the optical image (Fig. 4.8(c)). The two noise classes seem to be distributed a bit differently, as the red one is primarily concentrated around the area with blood vessels. Similar results are obtained in the second example MR sequence, only in this scan



(a) (b) (c)

Figure 4.8: Comparison of segmentations. a: Clustering by MSVC. b: difference in T2 weighted images. c: optical image of the extracted chamber.

MSVC divides the data in 4 clusters, which can be seen in Figure 4.9. In Figure 4.10 the result of the clustering is compared to the ΔT2 images and to an optical image. Not all of the blood vessels apparent in the optical image are detected in the ΔT2 image, hence the need for further refinement using a clustering method.

4.4 Conclusion

The proposed multiscale support vector clustering (MSVC) is demonstrated to give sensible results on a random data set and in a real clustering application, and this with the same parameters in spite of the fact that the two data sets are very different in scale, n and dimensionality. The presented method is rather fast and is virtually independent on the dimensionality if the problem.

The proposed clustering algorithm has only one parameter, which is the threshold for splitting clusters, and this parameter correlates strongly with the number of clusters (and their quality in terms of separation). We therefore believe that MSVC can be a very useful tool in many applications where it is possible to define a kernel.

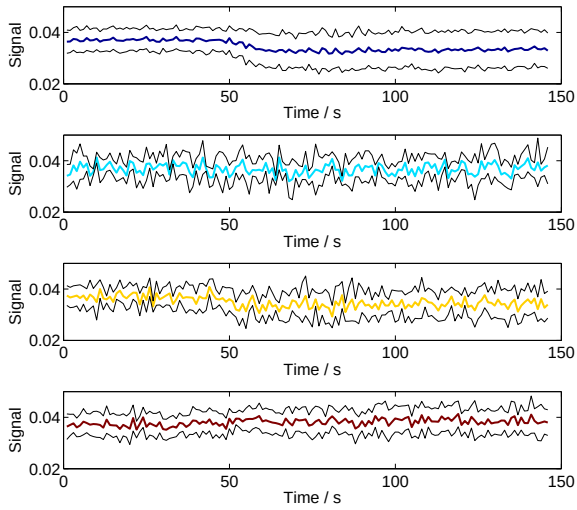
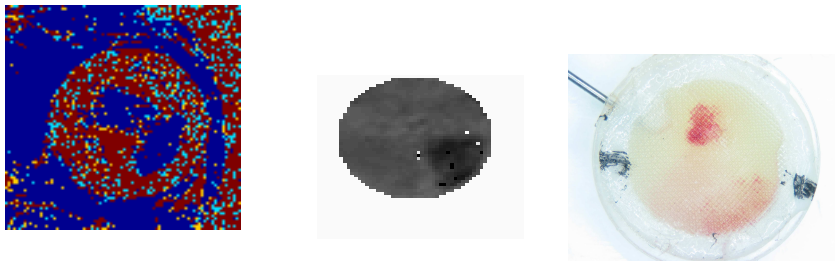


Figure 4.9: Curves showing the average of each of the three clusters $\pm$ one standard deviation. The bolus is only visible in one and thus blood related curve. The clustering is done on the sequence shown in Figure 4.4



(a) (b) (c)

Figure 4.10: Segmentation comparison. a:Clustering by MSVC. b: difference in T2 c: optical image of the extracted chamber

CHAPTER 5

Bayesian formulation of the Support Vector Domain Description

In the current work a Bayesian formulation of support vector domain description (SVDD) was developed having SVDD, as it was presented by Tax and Duin in 1999, as the maximum a posteriori solution. The introduced method is shown to deliver estimates of the probability and also a measure of the certainty of this estimate. Using the algorithm developed in [49], the entire regularization path can be efficiently calculated, and calculating the derived approximated evidence of the data is straightforward when the entire regularization path is available.

The method was tested on 1 and 2-dimensional artificial data sets, of which the 1-dimensional is included in the presentation for illustration. The tests seem promising considering the number of approximations involved, but the formulation still needs to prove its worth on real world problems.

5.1 Introduction

We are often faced with data of high dimensionality. Imaging devices with an intrinsic high number of variables are emerging for more and more applications, and in order to deal with this class of data, a whole series of data analysis tools have emerged. Often they are using the kernel trick, to create efficient algorithms dealing seamlessly with the high number of dimensions through inner products, while keeping flexibility for modelling distributions [106]. The support vector domain description (SVDD), introduced by Tax and Duin [101] in 1999, is a method for one-class labelling, which also falls in the aforementioned category. SVDD may be used for novelty detection as was done by Zhang et al. [115], for clustering, which has become a whole separate field of study [3, 110] or outlier detection [103].

To increase robustness Tax and Duin included a penalty term for negative examples in their support vector domain description [104]. Park et al. have used SVDD combined with geodesic distance for pattern denoising [79]. Seo and Ko used SVDD for face detection [91]. However, neither provided information on how to estimate the parameters. Guo et al. demonstrate how postprocessing the results of the SVDD could give a tighter fit, calculating a new boundary function, based on an average distance to the SVDD nearest boundary points [36]. This involves setting a decision threshold, which has to be set after empirical trials. Schölkopf et al. introduced a different one-class classifier, which finds the hyperplane that best separates the data from the origin. They show that for radial basis kernels, the data really lies on sphere, and finding the optimal hyperplane corresponds to finding a hypersphere which segments the part of the sphere, which is mainly populated by data, ie. yielding the same results as SVDD [88, 6].

In 2004, Sollich used Bayesian methods for analyzing the support vector machines [97]. This Bayesian analysis enabled him to estimate the evidence of the method, which may be used for model evaluation [69]. Several interesting methods for improving SVDD have been proposed, however a Bayesian calculation of evidence would still be of much use for estimating evidence of different sets of parameters. This is the motivation for the current work presented here. In 2000, Tipping introduced the *Relevance Vector Machine*, which gives a Bayesian interpretation of the support vector machine, and introduces a set of relevance parameters that choose the relevance of a vector for either regression or classification [105]. The relevance vector machine is, however, defined similar to the SVM problem and has a drawback in the significant computational cost associated with the optimization.

5.2 Support Vector Domain Description: A probabilistic interpretation

Suppose a data set $\mathcal{D}$ consisting of n samples $\boldsymbol{x}_i$ is given. Following the idea of SVDD introduced in [102], the data is analyzed by estimating an optimal enclosing hypersphere, containing most of the data examples, while minimizing the distance to outliers, and the radius of the sphere. In [95] it is demonstrated how the entire regularization path can be calculated efficiently, and using the same notation the optimization problem can be stated as

$$\min_{R,\boldsymbol{a}} \sum_{\boldsymbol{x}_i \in \mathcal{D}} \left(\|\boldsymbol{x}_i - \boldsymbol{a}\|^2 - R^2 \right)_+ + \lambda R^2 \tag{5.1}$$

where λ is a regularization parameter, R is the radius of the sphere, $\boldsymbol{a}$ is the center of the sphere, and $(\cdot)_+$ denotes the (reverse and shifted) hinge loss function given by

$$(x)_+ = \begin{cases} 0 & , \, x \leq 0 \\ x & , \, x > 0 \end{cases}$$

To allow the sphere to assume a more general shape, the data may be projected into an expanded basis, and by defining the inner product in the expanded basis $< \cdot, \cdot >$, expression (5.1) may be stated as the following minimization problem

$$\lambda R^2 + \sum_{\boldsymbol{x}_i \in \mathcal{D}} \left(\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_i) \rangle - 2 \langle \phi(\boldsymbol{x}_i), \boldsymbol{a} \rangle + \langle \boldsymbol{a}, \boldsymbol{a} \rangle - R^2 \right)_+ \tag{5.2}$$

where $\phi(\boldsymbol{x})$ is a transformation to a basis that may be expanded. To introduce a probabilistic interpretation of (5.2) it is natural to assume that it describes the negative log-posterior probability for the parameters $\boldsymbol{a}$ and R, given $\mathcal{D}$. The ordinary SVDD would then be interpreted as the maximum a posteriori (MAP) solution. The first term corresponds to a prior

$$Q(\boldsymbol{a}, R) \propto \exp\left(-\lambda R^2 - \frac{1}{2} \gamma^{-2} \langle \boldsymbol{a}, \boldsymbol{a} \rangle \right) \tag{5.3}$$

where the hyperparameter λ (the usual regularization parameter) controls the inverse variance of the radius R, γ are standard deviations, and taking the limit $\gamma \to \infty$ corresponds to a non-informative prior on $\boldsymbol{a}$, as is implicitly assumed in normal SVDD, where no restrictions are placed on the center of the sphere.

Since (5.2) is defined solely using inner products in the basis of the image of ϕ, it is preferable to introduce variables taking the values of the inner products $\Phi_i =< \boldsymbol{x}_i, \boldsymbol{a} >$. It is seen that these new parameters may be generated from a Gaussian process, and the covariance of these zero-mean parameters is

$$\mathbb{E}\left(\Phi_i \Phi_j\right) = \mathbb{E}\left(\langle \boldsymbol{x}_i, \boldsymbol{a} \rangle \cdot \langle \boldsymbol{x}_j, \boldsymbol{a} \rangle\right) = \gamma^2 \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \equiv \gamma^2 K_{ij}$$

since the elements of $\boldsymbol{a}$ are assumed to be independent and identically distributed with variance γ^2. The inner product $\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$ is seen to define the covariance function K_{ij} and it is clear why the kernel function K_{ij} is generally termed the covariance in [11]. If the analysis is limited to stationary kernels, ie. meaning all norms are constant, the expressions for the original problem statement (5.1) and prior (5.3) become trivially constant with respect to changes in $\boldsymbol{a}$. However, let $\cdot^\dagger$ be the matrix Penrose pseudo inverse operator, then in the finite dimensional case, when the dimension of the expanded basis is smaller than the number of samples and the rank of K is equal to the dimension of the (finite dimensional) expanded basis, the prior (5.3) may be written

$$Q(\boldsymbol{\Phi}, R) \propto \exp\left(-\lambda R^2 - \frac{1}{2}\gamma^{-2}\boldsymbol{\Phi}^T K^\dagger \boldsymbol{\Phi}\right) \tag{5.4}$$

where K denotes a (square) matrix with elements (K_{ij}), and $\boldsymbol{\Phi}$ is a vector with elements Φ_i. Otherwise, when data is not spanning the whole expanded basis, let $\{\boldsymbol{u}_j^*\}$ denote a orthonormal basis for the part of $\boldsymbol{a}$ not spanned by $\{\boldsymbol{x}_i\}$, let $\{a_i^*\}$ be the projections of $\boldsymbol{a}$ on these basis vectors, and let $\boldsymbol{a}^*$ denote a vector with elements a_i^*. Then it is easily seen that the MAP estimate for $\boldsymbol{a}^*$ is $\mathbf{0}$. In this case, when the data is not spanning the entire (expanded) feature space, the prior (5.4) could be corrected by adding a third term $-\frac{1}{2}\gamma^{-2} < \boldsymbol{a}^*, \boldsymbol{a}^* >$ in the exponent, however for simplicity of presentation this will be assumed not to be the case. More interestingly however, is to replace the original formulation of $\langle \boldsymbol{a}, \boldsymbol{a} \rangle$ with $\boldsymbol{\Phi}^T K^\dagger \boldsymbol{\Phi}$ as was done in (5.4), even if stationary kernels are used, noting that this is in fact a motivated reformulation of the original problem.

The second term in (5.2), which is data dependent, may be interpreted as a likelihood term given by

$$\begin{aligned} &Q(\mathcal{D}|\boldsymbol{a}, R) \\ &\qquad \propto \exp\left[-\sum_{\boldsymbol{x}_i \in \mathcal{D}} \left(\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_i) \rangle - 2\langle \phi(\boldsymbol{x}_i), \boldsymbol{a} \rangle + \langle \boldsymbol{a}, \boldsymbol{a} \rangle - R^2\right)_+\right] \end{aligned}$$

Applying the new parametrization of $\langle \boldsymbol{a}, \boldsymbol{a} \rangle$ yields the following likelihood function

$$\begin{aligned} &Q(\mathcal{D}|\boldsymbol{\Phi}, R) \\ &\qquad \propto \exp\left[-\sum_{\boldsymbol{x}_i \in \mathcal{D}} \left(K_{i,i} - 2\Phi_i + \boldsymbol{\Phi}^T K^\dagger \boldsymbol{\Phi} - R^2\right)_+\right] \end{aligned} \tag{5.5}$$

Aiming at using the method for outlier detection it is useful to introduce a binary variables y_i, which takes the value 1 if the point $\boldsymbol{x}_i$ is outside the sphere and 0 if it is inside, ie. when the value of the hinge loss in (5.5) is 0.

$$P(y_i = 1 \backslash y_i = 0 | \mathcal{D}, \boldsymbol{\Phi}, R) = \begin{cases} 0 \backslash 1 & , \; K_{i,i} - 2\Phi_i + \boldsymbol{\Phi}^T K^\dagger \boldsymbol{\Phi} - R^2 < 0 \\ 1 \backslash 0 & , \text{ else} \end{cases} \tag{5.6}$$

This trivial definition is already normalized.

The normalization factor of $Q(\mathcal{D}|\mathbf{\Phi}_i, R)$ depends in general on $\mathbf{\Phi}$ and R, so using $Q(\mathcal{D}|\mathbf{\Phi}, R)$ directly as stated in (5.5) would change the property that the SVDD is obtained as the MAP solution. To avoid this problem, the joint probability $P(\mathcal{D}, \mathbf{\Phi}, R)$ is written as

$$P(\mathcal{D}, \mathbf{\Phi}_i, R) = Q(\mathcal{D}|\mathbf{\Phi}, R)Q(\mathbf{\Phi}, R)/\mathcal{N}(\mathcal{D}) \; . \tag{5.7}$$

The shape of the posterior probability $P(\mathcal{D}|\mathbf{\Phi}, R) \propto Q(\mathcal{D}|\mathbf{\Phi}, R)Q(\mathbf{\Phi}, R)$ is by construction having the SVDD as the MAP solution. The normalization factor $\mathcal{N}(\mathcal{D})$ is chosen most simple to be $\mathcal{D}$ - independent, and it may be defined as

$$\begin{aligned}
\mathcal{N}(\mathcal{D}) = \mathcal{N} &= \int Q(\mathbf{\Phi}, R)N(\mathbf{\Phi}, R)d\mathbf{\Phi}dR \\
N(\mathbf{\Phi}, R) &= \sum_{\{\boldsymbol{x}_i\}} Q(\{\boldsymbol{x}_i\})Q(\mathcal{D}|\mathbf{\Phi}, R) \\
&= \sum_{\{\boldsymbol{x}_i\}} Q(\{\boldsymbol{x}_i\}) \exp\left[- \sum_{\boldsymbol{x}_i \in \mathcal{D}} \left(K_{i,i} - 2\Phi_i + \mathbf{\Phi}^T K^{\dagger} \mathbf{\Phi} - R^2 \right)_+ \right]
\end{aligned}$$

where $Q(\{\boldsymbol{x}_i\})$ is some prior on the obtained set of input vectors. Usually these vectors are assumed sampled independently from the input space. Using this definition sampling can be done by sampling a vector $\mathbf{\Phi}$ and a radius R from the Gaussian prior $Q(\mathbf{\Phi}, R)$, then sample a set $\{\boldsymbol{x}_i\}$ from $Q(\{\boldsymbol{x}_i\})$, and if a realization of a uniform random variable is smaller than $Q(\{\boldsymbol{x}_i\}|\mathbf{\Phi}, R)$, the sample is accepted. $Q(\{\boldsymbol{x}_i\}|\mathbf{\Phi}, R)$ is guaranteed to be smaller than 1, however it may be quite improbable for many realizations of $\mathbf{\Phi}$. In the case of a high number of samples, $\mathbf{\Phi}$ is sampled from a Gaussian process with one particular covariance structure, and later (ignoring the hinge loss) evaluated with a different covariance structure, which makes practical sampling hard in high dimensions.

The actual prior distribution is then given by

$$P(\mathbf{\Phi}, R) \propto Q(\mathbf{\Phi}, R)N(\mathbf{\Phi}, R) \; , \tag{5.8}$$

and the likelihood by

$$P(\mathcal{D}|\mathbf{\Phi}, R) \propto Q(\mathcal{D}|\mathbf{\Phi}, R)/N(\mathbf{\Phi}, R) \; , \tag{5.9}$$

Given the shape of the likelihood it is not so surprising that a the normalization depends on $\mathbf{\Phi}$ and R.

Using the introduced representation, the log-posterior can be maximized, which gives the usual SVDD result.

5.2.1 Maximizing the log-posterior

We maximize the log-posterior of the model

$$-\lambda R^2 - \frac{\gamma^{-2}}{2}\mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - \sum_i \left(-2\Phi_i + K_{i,i} + \mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - R^2\right)_+$$

Replacing the hinge loss by a set of values α_i where $\alpha_i = 1$ when the distance is bigger than zero, $\alpha_i = 0$ if it is negative, and $0 \leq \alpha \leq 1$ when the distance is zero (this process is equivalent to introducing slack variables and Lagrange multipliers as done in Chapter 3), gives the expression,

$$-\lambda R^2 - \frac{\gamma^{-2}}{2}\mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - \sum_i \alpha_i \left[-2\Phi_i + K_{i,i} + \mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - R^2\right] \tag{5.10}$$

By differentiating with respect to $\mathbf{\Phi}$, where $\boldsymbol{\alpha}$ conveniently represents a vector containing the elements (α_i) we get

$$\begin{aligned} 2\boldsymbol{\alpha} - \left[\gamma^{-2} + 2\sum_i \alpha_i\right] K^\dagger \mathbf{\Phi} &= \mathbf{0} \\ \Rightarrow \mathbf{\Phi}_{MAP} &= \frac{1}{\sum \alpha_i + \frac{1}{2}\gamma^{-2}} K\boldsymbol{\alpha} \,. \end{aligned}$$

Using this expression we arrive at the Lagrangian dual similar to the one described in [95], which has to be maximized with respect to $\{\alpha_i\}$, to achieve the MAP solution, supposing $\sum_i \alpha_i$ is known:

$$\sum_i \alpha_i K_{i,i} - \frac{1}{\sum \alpha_i + \frac{1}{2}\gamma^{-2}} \sum_{i,j} \alpha_i \alpha_j K_{i,j} + \left(\lambda - \sum_i \alpha_i\right) R^2 \tag{5.11}$$

where the sum $\sum_i \alpha_i = s$ is assumed fixed. By differentiating with respect to R^2 it is seen that

$$\sum_i \alpha_i = \lambda \,,$$

and substitution into (5.11) demonstrates how the problem is identical to the SVDD minimization problem (3.8), where λ determines $\sum_i \alpha_i$.

$$\sum_i \alpha_i K_{i,i} - \frac{1}{\lambda + \frac{1}{2}\gamma^{-2}} \sum_{i,j} \alpha_i \alpha_j K_{i,j}$$

The expression is noted to be increasing with increasing values of λ. If $K_{i,j}$ represent stationary kernels, e.g. the Gaussian kernel [32], the optimum α_{MAP} is independent of γ, and an obtained solution is thus the same as the one from normal SVDD.

5.3 Marginalization of parameters: calculating the evidence

The probabilistic formulation of the SVDD, introduced in Section 5.2, is enabling the calculation of the evidence of the data, given model and hyperparameters. In a true Bayesian treatment the hyperparameters λ and γ should be marginalized. With no prior on λ, the evidence of the assumed model is given by

$$P(D) = \int P(D|\lambda,\gamma)d\lambda d\gamma$$

$P(D|\lambda,\gamma)$ may be estimated using (5.7), and again the hyperparameters are implicitly assumed given for notational convenience.

$$P(D|\lambda,\gamma) = \int P(D|R,\mathbf{\Phi})P(R,\mathbf{\Phi})d\mathbf{\Phi}dR$$
$$\int P(D|R,\mathbf{\Phi})P(R)P(\mathbf{\Phi})d\mathbf{\Phi}dR$$

Returning to the expression for the MAP estimate (5.11) it is seen that $s = \sum_i \alpha_i$ alone is determining the MAP estimate of $\mathbf{\Phi}$. Remembering that the values $\{\alpha_i\}$ depend on the data likelihood and the argument of the hinge loss function, it is realized that the optimal value, given s, can only be obtained if R^2 corresponds to the MAP estimate of R^2. Let s be given by $s = \lambda$ then

$$P(D|\lambda,\gamma) = \int \exp\left[-\lambda R^2 - \frac{\gamma^{-2}}{2}\mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - \right.$$
$$\left. \sum_i \left(K_{i,i} - 2\Phi_i + \mathbf{\Phi}^T K^\dagger \mathbf{\Phi} - R^2\right)_+\right] / \mathcal{N} d\mathbf{\Phi} dR \qquad (5.12)$$

To estimate this integral, an approximation is made, where $\mathbf{\Phi}$ is developed around the MAP value given the sum $s = s(R^2)$. Let $\alpha_{MAP,i}$ be the MAP values of the parameters α_i, and define $\mathbf{\Phi} = \mathbf{\Phi}_{MAP} + \Delta\mathbf{\Phi}$, where $\Delta\mathbf{\Phi}$ is the perturbation of the MAP estimate, then

$$P(D|\lambda,\gamma) = \int \exp\left[-\lambda R^2 - \frac{\gamma^{-2}}{2}\Delta\mathbf{\Phi}^T K^\dagger \Delta\mathbf{\Phi} - \right.$$
$$\left. \sum_i \left(d_i - 2\left(\Delta\Phi_i - \frac{\sum_i \alpha_{MAP,i}\Delta\Phi_i}{\sum \alpha_{MAP,i}}\right) + \Delta\mathbf{\Phi}^T K^\dagger \Delta\mathbf{\Phi}\right)_+\right] / \mathcal{N} d\mathbf{\Phi} dR \qquad (5.13)$$

is approximated using a modified version of Laplace's method [69]. However, inspecting the argument of the hinge loss, it is seen that for small perturbations $\Delta\mathbf{\Phi}$, the sum is dominated by the constant terms, and for increasing perturbations, the quadratic term

becomes dominant. Therefore, the sum over all data points is assigned the values n and $\sum_i \alpha_i$ respectively. The linear terms in the sum are seen to approximately cancel out except for skewness induced by the hinge loss function. The approximated evidence becomes

$$\begin{aligned}
P(D|\lambda,\gamma) \approx & \int \exp\left[-\lambda R^2\right] \cdot \exp\left[-\frac{1}{2}\Delta\boldsymbol{\Phi}^T\left(2nK^\dagger + \gamma^{-2}I_n\right)\Delta\boldsymbol{\Phi}\right] \\
& \cdot \exp\left[-\sum_i \alpha_{MAP,i}K_{i,i} + \boldsymbol{\alpha}_{MAP}^T K\boldsymbol{\alpha}_{MAP} + R^2\sum_i \alpha_{MAP,i}\right] / \mathcal{N} d\Delta dR\boldsymbol{\Phi} \\
= & \frac{(2\pi)^{n/2}}{\mathcal{N}\sqrt{|2nK^\dagger + \gamma^{-2}I_n|}} \int \exp\left[R^2\sum_i \alpha_{MAP,i}\right] \\
& \cdot \exp\left[-\lambda R^2 - \sum_i \alpha_{MAP,i}K_{i,i} + \boldsymbol{\alpha}_{MAP}^T K\boldsymbol{\alpha}_{MAP}\right] dR\,. \qquad (5.14)
\end{aligned}$$

By using the entire regularization path of the SVDD, this integral can be computed to an approximation of smaller order than the previous approximations, by calculating the trapezoidal sum along the regularization path. This is due to the fact that the radius R is decreasing monotonic as a function of the regularization parameter λ, as noted in Section 5.2.1. In the Appendix 5.6.1 details on calculating R are included.

Unfortunately, there seems to be no straightforward way to estimate the normalization constant $\mathcal{N}$. The sampling procedure discussed earlier is too cumbersome for higher dimensions, and the integral is hard to approximate for the general case. It has been argued by Opper and later by Sollich that the Bayesian treatment still may give sensible results, even when the evidence is not properly normalized [97, 77]. However, it is the view and experience of the author that at least obtaining a reasonable approximation would be much preferable, if one wants to compare different covariance functions (or the implicitly given parameters of these, as in Chapter 4).

5.3.1 The expected value of y_i

The output that is actually expected to be of interest in the analysis is the individual classifications of the input vectors $\boldsymbol{x}_i$ as either inliers or outliers. In (5.6) the variables y_i were defined, which make this distinction. These variables depend on the parameters (and the hyperparameters) as well as on the input data (more specifically on the covariance of the input). However, using (5.14) the expected value (considering the parameters as stochastic variables) may be calculated without having to find the nor-

malization factor $\mathcal{N}$. The hyperparameters are implicitly assumed to be given by

$$\begin{aligned}\mathbb{E}(y_i) =& 0 \cdot P(y_i = 0|\mathcal{D}) + 1 \cdot P(y_i = 1|\mathcal{D}) \\ =& \int 1 \cdot P(y_i = 1|\mathcal{D}, \mathbf{\Phi}, R) P(\mathbf{\Phi}, R|\mathcal{D}) d\mathbf{\Phi} dR \\ =& \frac{\int 1 \cdot P(y_i = 1|\mathcal{D}, \mathbf{\Phi}, R) P(\mathcal{D}|\mathbf{\Phi}, R) P(\mathbf{\Phi}, R) d\mathbf{\Phi} dR}{P(\mathcal{D})} .\end{aligned} \tag{5.15}$$

A scaled version of $P(\mathcal{D}|\mathbf{\Phi}, R)$ may be used, since it is normalized with the same scaled version when calculating $P(\mathcal{D})$ using (5.14). Given the definition of the expectation, and the variable y_i it is seen that the expectation can actually be perceived as the probability for the point $\boldsymbol{x}_i$ to be an outlier. Following the same arguments we may calculate the variance of y_i

$$\begin{aligned}&\mathbb{E}((y_i - \mathbb{E}(y_i))^2) \\ &\qquad = (0 - \mathbb{E}(y_i))^2 \cdot P(y_i = 0|\mathcal{D}) + (1 - \mathbb{E}(y_i))^2 \cdot P(y_i = 1|\mathcal{D})\end{aligned}$$

However, for binary variables the variance is given by $\mathbb{E}((y_i - \mathbb{E}(y_i))^2) = \mathbb{E}(y_i) - \mathbb{E}(y_i)^2$.

5.4 Example

A small 1-dimensional test example was generated to demonstrate the method, consisting of two clusters of Gaussian randomly sampled points with variances 0.75 and 1.05, and centered with a distance of 6, along with 20 Gaussian distributed noise observations centered at the mean with a variance 35. The example is analyzed with the ultimate goal of finding the mean classification of each point, as well as the variance of the estimate. A Gaussian kernel, with a kernel width of 10 is chosen, and illustrated along with the sampled points in Figure 5.1.

The regularization parameter λ was set at 20, coinciding with the number of true outliers, ie. the number of noisy observations added to the example. The expected value, and variance of y_i for each of the points was calculated as described in Section 5.3.1, only the distribution with respect to $\Delta\mathbf{\Phi}$ is assumed to be dense around the MAP value $\mathbf{\Phi}^*$ such that the whole probability may be assumed to lie at this value. The results are visualized in Figure 5.2.

The proposed method is seen to identify the two clusters, however with some errors due to the overlapping groups, and, maybe, also the need for support vectors to define a cluster. The variance of the estimates is particularly interesting, and a plot of it can be seen in Figure 5.3.

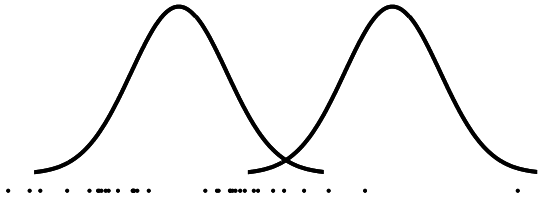


Figure 5.1: The distribution of the 34 1D points is illustrated, as well as the size of the kernel used .

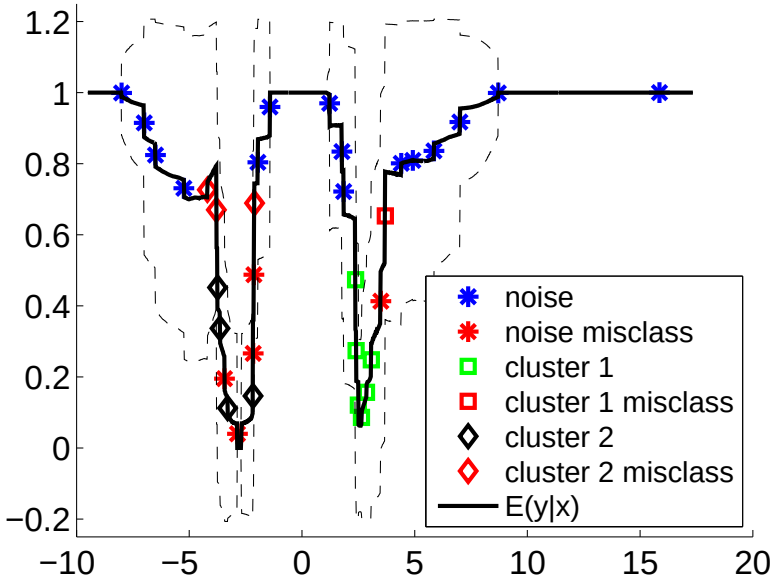


Figure 5.2: Illustration of classification result of the test example, along with the decision function on the interval, where the dashed lines show the expected value $\pm$ one standard deviation.

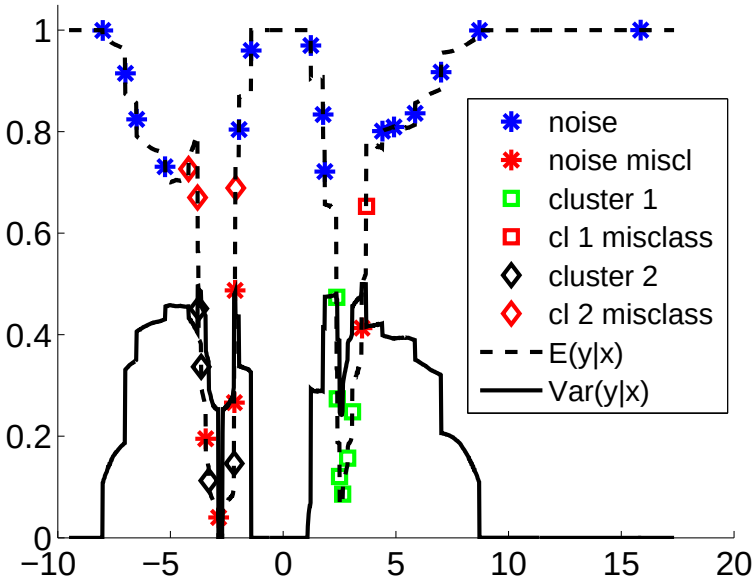


Figure 5.3: Illustration of the variance of the classification estimates in Figure 5.2. The variance is seen to be bigger on the boundaries between clusters and outliers.

It is evident from the variance plots, that the variance of the estimates is bigger on the boundaries between the clusters and the outliers. This is a natural consequence of the definition of the variance, however the width of the peaks is not. The wide peaks can be perceived to reflect the fact that it is harder to tell (given the scale) if the left outlier points are actually inliers or outliers. It is interesting to note, that the variance is also comparably small in the cluster centers, where the distance to support vectors is bigger. This may be seen as an argument for the support vector idea, in the sense that modelling the boundary is the most important for (one-class) classification. The distribution of the conditional probability given R is shown in Figure 5.4. The distribution is seen to be lightly peaked, with a small favor to smaller radii, which corresponds well with the choice of regularization parameter λ.

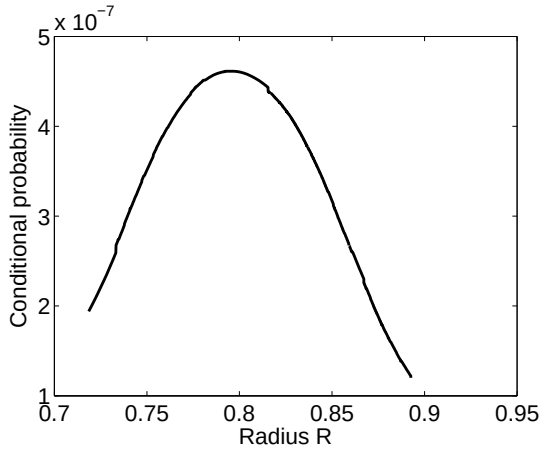


Figure 5.4: Relation between the conditional evidence for the data given the radius R, ie. with $\mathbf{\Phi}$ marginalized (the evidence has not been properly normalized). The λ value of 20 favors smaller radii, since it a priori assumes a high number of outliers.

5.5 Discussion and conclusion

In the current work a Bayesian formulation of SVDD was developed having the SVDD presented by Tax and Duin [101] as the maximum a posteriori solution. In Section 5.4, the introduced method was shown to deliver estimates of the probability, and also a measure of the certainty of this estimate. Using the algorithm developed in Chapter 3 the entire regularization path can be efficiently calculated, and calculating the derived approximated evidence (5.14) is straightforward when the entire regularization path is available. Marginalizing λ may be quite straightforward without adding too much

computational load, but estimating the normalization coefficient remains an open problem. The method was tested on 1 and 2 dimensional artificial data sets, where the 1-dimensional has been presented for visualization. The results seem promising considering the approximations involved and the small complexity, but the formulation still needs to prove its worth on real world problems.

5.6 APPENDIX

5.6.1 Calculating R

R is clearly defined when the value of $\sum_i \alpha_i$ is at an event, and in between, the value can be calculated by using the standard distance function on boundary points defined in (3.11) in Chapter 3.

$$R^2 = k_b - \frac{2}{\sum_i \alpha_i} \cdot \sum_i K_{b,i}\alpha_i + \frac{1}{\left(\sum_i \alpha_i\right)^2} \cdot \boldsymbol{\alpha}^T K \boldsymbol{\alpha}$$

Making the substitution from the linear regularization path (3.13), representing $\boldsymbol{\alpha} = \boldsymbol{\alpha}^j + t\boldsymbol{p}$, where $\boldsymbol{\alpha}^j$ is the value of $\boldsymbol{\alpha}$ from the previous event, and $\boldsymbol{p}$ is the slope of the (linear) change in $\boldsymbol{\alpha}$, and t is a parametrization of the line. Making the substitution, the result is

$$R^2 = \\ k_b - \frac{2}{\sum_i \alpha_i^j + t} \cdot \sum_i K_{b,i}(\alpha_i^j + tp_i) + \frac{1}{\left(\sum_i \alpha_i^j + t\right)^2} \cdot (\boldsymbol{\alpha}^j + t\boldsymbol{p})^T K (\boldsymbol{\alpha}^j + t\boldsymbol{p})$$

In the case when the algorithm is initialized, for no regularization, all points are inliers, ie. $\boldsymbol{\alpha} = \mathbf{0}$, and the radius assumes the following fixed value until the next event

$$R^2 = k_b - 2\sum_i K_{b,i}p_i + \boldsymbol{p}^T K \boldsymbol{p} \tag{5.16}$$

CHAPTER 6

Discussion and conclusion

In Chapter 3, an efficient algorithm for calculating the entire regularization path of the support vector domain description (SVDD) was presented. The ability to calculate the entire path with a complexity in the same order as solving the original quadratic problems gives inspiration to utilize the extra information available from the entire path. Not much effort in the literature on SVDD has been put into determining the optimal value of the regularization, or possibly an ensemble of values, and some efforts appear ad hoc [3]. In Chapter 4, a method for hierarchical support vector clustering, based on information from the entire regularization path, and multiple Gaussian kernels is described. This method can be considered an extension of a previously developed method, which is also empirically shown to give good results on real world data [40].

Testing the methods on more data would be interesting in the future to draw more conclusions on the nature of the support vector domain description for clustering. However, this would also be very interesting to see combined with the analysis presented in Chapter 5, where Bayesian methods are applied in the attempt to draw direct statistical conclusions from the SVDD analysis.

Part II

Parametrization of deformations

CHAPTER 7

Introduction to parameterizations of deformations

Computations performed on a computer need by nature to be parameterized at some point to a finite dimensional space. This may be done in different ways, for instance by a discretize-optimize approach or an optimize-discretize approach [73]. In image registration, the Cubic B-spline kernels used by Rueckert et al. [85] and the cosine kernels presented by Cootes et al. [16] are both examples of discretizations that restrict the optimization to the finite dimensional space of the basis. The parameters are in both cases controlling the deformation field in x,y and z directions, where the support of the control point influence is given by the associated basis function, and the deformation field is given as a linear combination of the basis functions, where the linear coefficients are the control points.

For some applications, however, it has been shown to be favorable to choose a different parametrization to increase the stability of the optimization. In electrostatics and fluid flow estimation parameterizing by the divergence and the curl has this property [57, 60]. In Chapter 8 an image registration parametrization is introduced, which defines the k parameters in $\mathbb{R}^k$, while guaranteeing that the resulting deformation field is diffeomorphic [45]. In Chapter 9 deformation potentials are presented, where the curl and the gradient of the potentials are controlling the deformation fields [42]. This

representation is shown to give an intuitive interpretation of the linear elastic regularization.

CHAPTER 8

Diffeomorphic Statistical Deformation Models

Michael S. Hansen, Mads F. Hansen and Rasmus Larsen [1]

Abstract

In this paper we present a new method for constructing diffeomorphic statistical deformation models in arbitrary dimensional images with a nonlinear generative model and a linear parameter space.

Our deformation model is a modified version of the diffeomorphic model introduced by Cootes et al. The modifications ensure that no boundary restriction has to be enforced on the parameter space to prevent folds or tears in the deformation field.

For straightforward statistical analysis, principal component analysis and sparse methods, we assume that the parameters for a class of deformations lie on a linear manifold and that the distance between two deformations are given by the metric introduced by the L_2-norm in the parameter space. The chosen L_2-norm is shown to have a clear and intuitive interpretation on the usual nonlinear manifold.

Our model is validated on a set of MR images of corpus callosum with ground truth in form of manual expert annotations, and compared to Cootes's model.

[1] This paper was presented at the NRTL workshop, which was part of the International Conference on Computer Vision 2007 [45].

We anticipate applications in unconstrained diffeomorphic synthesis of images, e.g. for tracking, segmentation, registration or classification purposes.

8.1 Introduction

Registration is the problem of establishing correspondence between points in different images. It has been used for building models of variation in groups of images for several years. Cootes et al. proposed the very successful active appearance models in 1998 [15], which, once trained, can establish correspondence between points in the model and the images using a piecewise affine mapping. Rueckert et al. presented a statistical deformation model based on registrations of an atlas to the images of the group [85]. Joshi et al. demonstrate how to construct an unbiased atlas from a population [59], and Cootes et al. presented a guaranteed diffeomorphic shape model [16] by using smooth kernels for interpolating a warp field and putting restrictions on the variation of the parameters. Vester-Christensen et al. have presented an accelerated version of this algorithm [108], which is based on the inverse compositional method by Baker et al., which we have also made extensive use of in the presented work [1].

8.2 Methods

We define image registration as the identification of correspondence between positions in images. In the current work we address problems where the correspondences can be represented by a diffeomorphic function $f \in \mathcal{H}$, where $\mathcal{H}$ denotes the infinite dimensional group of diffeomorphisms on $\mathbb{R}^N$. The mapping from one image to the other is differentiable and the inverse exists and is also differentiable. Popular speaking this limits the problem of registration to the problem of finding smooth warps without folds or tears. More precisely this is fulfilled, when the Jacobean of the warp field is positive and well defined.

In the statistical analysis of the warp functions we are interested in estimating an unbiased atlas of the structures we are registering. We identify such an atlas as the groupwise maximizer of similarity between the atlas R and the deformed images I_i, while minimizing the deformation fields ϕ_i.

$$[\phi_i,\ \hat{R}] = \min_{\phi_i, \hat{R}} \sum_i \mathcal{S}[\hat{R}, I_i \circ \phi_i] + \alpha \mathcal{D}(\phi_i)^2 \ . \tag{8.1}$$

where $\mathcal{S}$ denotes the similarity measure and $\mathcal{D}(\phi)$ denotes the regularization term, introduced to regularize the warp ϕ further than just restricting it to the space of the parameters, and α is the regularization parameter.

8.2.1 Parameterized diffeomorphisms

Fletcher et al. have investigated geodesic curves on the nonlinear manifolds of the parameters of the M-reps parameterization [28]. Most of the current statistical analysis, however, is based on the assumption that the data is located on a linear manifold with the Euclidean metric, e.g. principal component analysis (PCA) and independent component analysis (ICA), which have nice properties as analytical tools. This is our motivation for introducing a function $\mathcal{G}$ which identifies $\mathbb{R}^M$ with a (hopefully interesting) subset of diffeomorphisms.

Let $\mathcal{H}(\mathbb{R}^N)$ denote the set of diffeomorphisms $(f : \mathbb{R}^N \rightarrow \mathbb{R}^N)$. Now let $\mathcal{G}$ be a bijective mapping:

$$\mathcal{G} : \mathbb{R}^M \rightarrow \mathcal{H}_t \ . \tag{8.2}$$

where $\mathcal{H}_t = \mathcal{G}(\mathbb{R}^M) \subset \mathcal{H}$. We let $\mathcal{H}_t$ inherit the Euclidean metric from the parameter space $\mathbb{R}^M$

$$\begin{aligned} d(\mathcal{G}(t_1), \mathcal{G}(t_2)) \equiv d(t_1, t_2) = \|t_1 - t_2\|_2 \ , \\ t_1, t_2 \in \mathbb{R}^M \text{ and } \mathcal{G}(t_1), \mathcal{G}(t_2) \in \mathcal{H}_t \ , \end{aligned} \tag{8.3}$$

from which we conclude that $\mathcal{G}$ is a homeomorphism, and that the spaces $\mathcal{H}_t = \mathcal{G}(\mathbb{R}^M)$ and $\mathbb{R}^M$ are topologically equivalent. To conclude it can be observed that the defined metric on the space of parameterized warps is the L_2 norm on $\mathbb{R}^M$ as intended.

8.2.1.1 Composition of warps

The composition of more diffeomorphisms is diffeomorphic, which is a very important property of diffeomorphisms in the present context.

$$\begin{aligned} f_i \in \mathcal{H} \ , \ i \in \{1, 2, \ldots, n\} \\ \phi = f_n \circ f_{n-1} \circ \ldots \circ f_1 \ \Rightarrow \ \phi \in \mathcal{H} \end{aligned} \tag{8.4}$$

This allows for the construction of diffeomorphisms of higher complexity by the composition of several simpler warps. We shall assume we are dealing with parameterized warp functions, and our statistical analysis of warps can be reduced to the analysis of the warp parameters, in line with (8.3). For all images in our set the warp parameters shall warp from our *reference*, R, into the current *target*, I. In order to be able to compare parameters from different warp compositions it is evident that all our parameters exist in the same space. This is achieved by ensuring that all warps f_i in a composition warp from the reference coordinate system[16].

8.2.1.2 Grid based diffeomorphisms

Several grid based representations of diffeomorphisms have been presented and they are commonly used at different levels of detail and composed succeedingly [16, 73, 85]. A general trait of the grid methods is that they manipulate the parameters of the functions describing the diffeomorphism, and that the functions have a local support in the image, either as points defined in the image or as basis functions with support around a control point. Often this parameterization of the grid is linear in the parameters and this obviously imposes some restrictions on the parameters to produce diffeomorphic warps. Cootes et al. specify a cut-off at displacements larger than $\frac{1}{\pi}$ of the cosine based kernel [16] and Lee et al. find a threshold bound on the B-spline parameters to secure that the B-spline based warp function is diffeomorphic [66].

8.2.1.3 A proposed $\mathcal{G}$

Let $\mathcal{F}$ be the function mapping from a real parameter space $\mathbb{R}^M$ into the space of functions from $\mathbb{R}^N$ to $\mathbb{R}^N$, e.g. in case of the B-spline warps, $\mathcal{F}$ maps from the parameter space into the space of N-dimensional B-spline functions $f : \mathbb{R}^N$ to $\mathbb{R}^N$, the image of $\mathcal{F}$, $\mathcal{K}$ can be shown to contain functions that are not diffeomorphic.

As discussed in the previous section there can for some parameterized warps be specified a threshold such that $\mathcal{P} =]-\tau_1, \tau_1[\times \cdots \times]-\tau_M, \tau_M[$ and $\mathcal{F} : \mathcal{P} \rightarrow \mathcal{H}_t$, where $\mathcal{H}_t \equiv \mathcal{F}(\mathcal{P}) \subset \mathcal{H}$. In the current study we have investigated the use of a function $g : \mathbb{R}^M \rightarrow \mathcal{P}$, that is, a bounded monotonic injective function into the space of thresholded displacement parameters. Constructing $\mathcal{G} = \mathcal{F} \circ g$, where $\mathcal{G} : \mathbb{R}^M \rightarrow \mathcal{H}_t$ gives us the desired function $\mathcal{G}$, namely a homeomorphic mapping from the parameter space $\mathbb{R}^M$ into the space of diffeomorphisms. As an example of the function g we have chosen a set of hyperbolic tangent function, because the range where it is close to linear is large. The composed mapping $\mathcal{G}$ and the different ranges are illustrated in Figure 8.1.

We define g coordinate-wise by

$$\boldsymbol{g} = \{g_1, ..., g_M\} \text{ where } g_i : R \rightarrow]-\tau_i, \tau_i[$$

$$s_i = g_i(t_i) = \tau_i \tanh a_i t_i \text{ , for } \quad i \in \{1, ..., M\} \tag{8.5}$$

where τ_i are the threshold parameters reducing the displacement parameter space of the warp to $\mathcal{P} \subset \mathbb{R}^M$, $\boldsymbol{s} = \{s_1, ..., s_M\} \in \mathcal{P}$ are the displacement parameters and a_i are constants ensuring that the impact of each t_i is of the same order of magnitude.

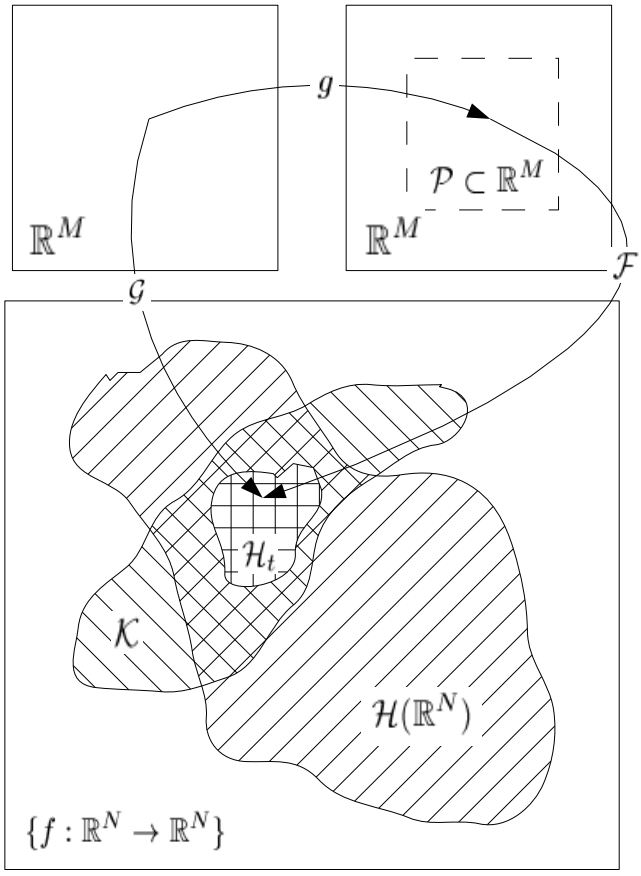


Figure 8.1: Illustration of the mapping $\mathcal{G}$ from $\mathbb{R}^N$ to $\mathcal{G}$, along with our proposed composed mapping $\mathcal{G} = \mathcal{F} \circ g$

.

8.2.1.4 Properties of the g mapping

Before we continue with an empirical validation of our proposed mapping we will make some theoretical considerations over the choice of homeomorphic mapping g. For small values $t \in \mathbb{R}^M$ the L_2 norm in $\mathbb{R}^M$ is equivalent to a scaled L_2 norm in $g(\mathbb{R}^M)$ to a first order. In other words, relating this to diffeomorphic warps, for small deformations the defined norm is equivalent to the usual metric applied in analysis of the warp fields [16, 85].

8.2.1.5 The parameter distribution

We believe that the distribution of the parameters is well described by normal distribution, and we will show what distribution this describes in the displacement parameter space of the warp function. Let f_{t_i} be the marginal distribution of the parameter t_i and f_{g_i} be the marginal distribution of the warp parameter $s_i = g_i(t_i)$, then

$$f_i(t_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{t_i^2}{2\sigma_i^2}} \tag{8.6}$$

$$f_{g_i}(s_i) = \frac{1}{2a_i \cdot \tau_i \sqrt{2\pi\sigma_i^2}} \left(e^{-\frac{g^{-1}(s_i)^2}{2\sigma_i^2}} + \frac{e^{\frac{\mu_i^2}{2\sigma_i^2}}}{2} e^{-\frac{(g^{-1}(s_i)-\mu_i)^2}{2\sigma_i^2}} + \frac{e^{\frac{\mu_i^2}{2\sigma_i^2}}}{2} e^{-\frac{(g^{-1}(s_i)+\mu_i)^2}{2\sigma_i^2}} \right) \tag{8.7}$$

where $\mu_i = \frac{\sigma_i^2 a_i}{2}$ and this distribution is seen to be the composition of three Gaussian distributions scaled by g^{-1}. For small μ_i this is approaching the Gaussian distribution which is often the distribution for the warp parameters in the small deformation domain and for μ_i big the two μ_i displaced distributions dominate, and we observe a high concentration of parameters around the threshold τ_i. In the presence of strong deformations this also what we expect when imposing a threshold on the warp deformation parameters. Based on these considerations we expect an M-dimensional normal distribution of our parameters to be well suited for modelling the distributions of the observed deformations.

8.2.1.6 Statistical deformation model

In the previous section we argued that the expected distribution of warps could be modelled as an M-dimensional normal distribution. If this is the case PCA is known to

be the optimal choice of analysis tool for creating a compact model of the observations, and is therefore the method of choice in the current implementation.

8.3 Implementation

To validate our approach for construction of diffeomorphic deformation model we have adapted the grid based diffeomorphisms by Cootes [16] with our g mapping. These diffeomorphisms can be viewed as an extension to standard linear interpolation, where the interpolation coefficients are transformed by a suitable kernel $k(r)$ which ensures smoothness across the grid boundaries. The displacement of a 2D point $\boldsymbol{x} \in \mathbb{R}^2$ is given by

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{d}) &= \sum_{m=0}^{1}\sum_{n=0}^{1} k_n(v)k_m(w)\boldsymbol{d}_{i+n,j+m} \\ &= \sum_{m=0}^{1}\sum_{n=0}^{1} a_{i+m,j+n}(\boldsymbol{x})\boldsymbol{d}_{i+n,j+m} \qquad (8.8) \\ &= \begin{bmatrix} \boldsymbol{a}(\boldsymbol{x})^\top & \mathbf{0} \\ \mathbf{0} & \boldsymbol{a}(\boldsymbol{x})^\top \end{bmatrix} \boldsymbol{d} \qquad (8.9) \end{aligned}$$

where $k_0(r) = k(r)$, $k_1(r) = 1 - k(r)$, i and j is the local indices of the neighboring grid points, v and w are relative positions of x in the neighborhood and $\boldsymbol{d}$ and $\boldsymbol{d}_{i,j}$ are all the displacements and the displacement of the (i, j)-node, respectively. By substituting the displacements $\boldsymbol{d}$ with the g mapping with a suitable threshold τ, this deformation model will no longer be able to generate non-diffeomorphisms. In the present example using the Cootes kernel, $\tau = 1/\pi$.

For notational simplicity the displacement in the ith direction will represented by

$$u_i(\boldsymbol{x}, \boldsymbol{t}_i) = \boldsymbol{a}(\boldsymbol{x})^\top \boldsymbol{g}_\tau(\boldsymbol{t}_i), \qquad (8.10)$$

and the warp function is written in the form

$$\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}) = \boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{t}). \qquad (8.11)$$

8.3.1 Image registration

To drive the registration between a reference image R and a target image I we apply the sum-of-squared-differences (SSD) as our similarity measure and the regularization term is given by $\mathcal{D}(\phi) = d(e, \phi) = \|t\|_2$, where e is the identity map corresponding to

$\boldsymbol{t} = 0$. The SSD comparison leads us to calculate the reference image as the arithmetic mean of the warped target images, as this is the optimum SSD solution to (8.1) [59].

$$F(\boldsymbol{t}) = \frac{1}{2}\sum_{\boldsymbol{x}}(R(\boldsymbol{x}) - I(\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{t}))^2 + \alpha\|\boldsymbol{t}\|_2^2 \tag{8.12}$$

$$= \frac{1}{2}\sum_{\boldsymbol{x}} E^2(\boldsymbol{x},\boldsymbol{t}) + \alpha\|\boldsymbol{t}\|_2^2. \tag{8.13}$$

To achieve a fast optimization we apply the inverse compositional optimization approach by Baker et al. [1] to the cost function. Thus, we obtain a minimum by iteratively minimizing

$$F_{ic}(\boldsymbol{t}) = \frac{1}{2}\sum_{\boldsymbol{x}}(R(\boldsymbol{\varphi}(\boldsymbol{x},\Delta\boldsymbol{t})) - I(\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{t}))^2 + \alpha\|\boldsymbol{t} - \frac{\partial \boldsymbol{t}'}{\partial \Delta \boldsymbol{t}}\Delta\boldsymbol{t}\|^2 \tag{8.14}$$

with respect to $\Delta\boldsymbol{t}$ and updating $\boldsymbol{t}$ according to

$$\boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{t}') \leftarrow \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{t}) \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{x},\Delta\boldsymbol{t}). \tag{8.15}$$

In Appendix 8.7.2 it is shown how $\boldsymbol{t}'$ is derived from (8.15).

By performing a first-order Taylor-expansion on $R(\boldsymbol{\varphi}(\boldsymbol{x},\Delta\boldsymbol{t})$ around $\boldsymbol{x}$ in (8.14), taking the derivatives wrt. $\Delta\boldsymbol{t}$ and setting them equal to zero we get

$$\Delta\boldsymbol{t} = \boldsymbol{H}^-1\left[\sum_{\boldsymbol{x}} \boldsymbol{SD}(\boldsymbol{x})^\top E(\boldsymbol{x},\boldsymbol{t}) + \alpha\frac{\partial \boldsymbol{t}'}{\partial \Delta \boldsymbol{t}}^\top \boldsymbol{t}\right] \tag{8.16}$$

where

$$\boldsymbol{SD}(\boldsymbol{x}) = \nabla R(\boldsymbol{x})\frac{\partial \boldsymbol{\varphi}(\boldsymbol{x},\boldsymbol{0})}{\partial \boldsymbol{t}} \tag{8.17}$$

and

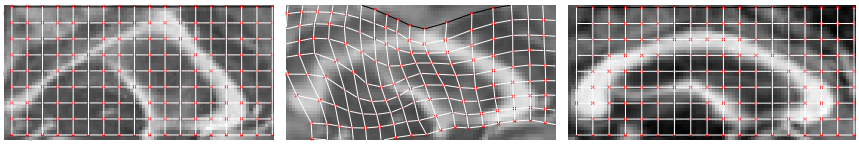
$$\boldsymbol{H} = \sum_{\boldsymbol{x}} \boldsymbol{SD}(\boldsymbol{x})^\top \boldsymbol{SD}(\boldsymbol{x}) + \alpha\left[\frac{\partial \boldsymbol{t}'}{\partial \Delta \boldsymbol{t}}\right]^\top\left[\frac{\partial \boldsymbol{t}'}{\partial \Delta \boldsymbol{t}}\right]. \tag{8.18}$$

The advantages with this inverse compositional approach is that $\boldsymbol{SD}(\boldsymbol{x})$ can be precomputed as it is not dependent on $\boldsymbol{t}$.

8.4 Validation: corpus callosum model

To demonstrate our approach we have created a deformation model of the Corpus Callosum from 62 two dimensional MR images of the mid-sagittal cross-section of the corpus callosum brain structure. This data set is part of the LADIS (Leukoaraiosis and DISability) study [78], a pan-European study involving 12 hospitals and more than 700 patients. Furthermore, each corpus callosum have manually been annotated with 72 landmarks by a clinician, which we will later use for validation.

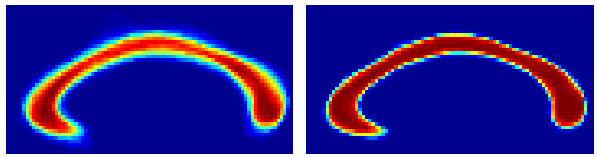
Prior to the non-rigid registration a rigid registration was performed to filter out non-anatomical variation. This was achieved by performing Procrustes analysis on the sets of annotation. After the rigid registration an initial reference was created by computing the mean image of the rigid registered images. All corpus callosum images were then non-rigidly registered to the reference, and a new reference was computed by averaging. This was done multiple times until the reference stabilized. For the non-rigid registration the cosine kernel $k(r) = 0.5(1 + cos(\pi r))$ was applied [16]. The non-rigid warps were modelled by composing three grid based diffeomorphisms in a fine-to-coarse manner. The dimensions of the applied grids were 5×4, 10×8 and 20×16. The non-rigid registrations were carried out in coarse to fine order. After each level φ_i of the warp was estimated the target image was updated by warping the target image back into the reference coordinate frame by $T_{n+1}(x) = T_n(\varphi(x))$. This was done to ensure that different parameters from different warps could be compared [16]. a_i of the g mapping was set proportional to the inverse of the squared grid node distance because the grid was 2 dimensional. The image registration was validated using the Dice measure, which is twice the intersecting area between the ground truth shape outline of the warped image and the outline of the reference shape divided by the total area inside the two outlines. The ground truth was obtained from the expert annotations. The Dice measure was 0.884 ± 0.048. In Fig. 8.2 we show an example of a typical registration of an image. In Fig. 8.3 the cumulative overlap of the aligned corpus callosum shapes before and after a rigid registration is illustrated, showing a clear improvement in correspondences between the shapes.



(a) Template image (b) Warped template (c) Reference image

Figure 8.2: Registration of an image to the reference.

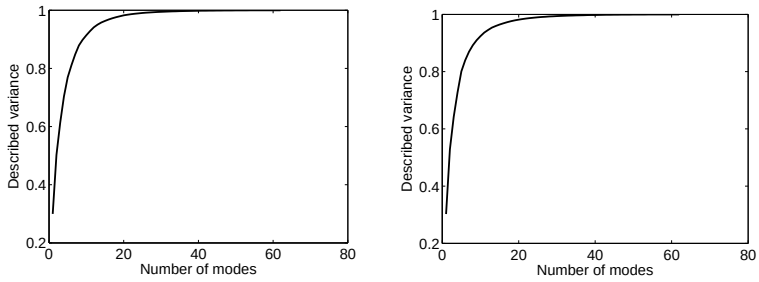
To create a compact deformation model, PCA was applied to the parameters after the groupwise registration of the images. 13 modes of variation could describe 95 % of the



(a) Before nonrigid registration (b) After nonrigid registration

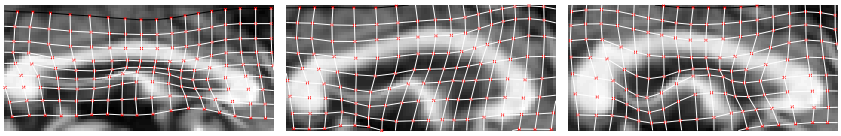
Figure 8.3: Cumulative overlap of the aligned corpus callosum shapes before and after a rigid registration

observed variation in the population as observed in Figure 8.4, and the first three modes are illustrated in Fig. 8.5. The first mode of variation is seen to be related to a vertical stretch and in particular to the size of the septum pellucidum (the dark area between the bright corpus callosum and the bright Fornix), the second mode is related to the kink of the corpus callosum and the thickness of the structure and the same goes for the third mode but with a different bending of the Fornix. Rueckert et al. have also analyzed the corpus callosum and they found modes quite similar to the ones found in the current study [85]. For comparison we applied a regularized version of Cootes' algorithm to the same problem, and constructed a similar PCA model of the variation. The variance of the modes is nearly identical, as shown in Figure 8.4 and the obtained Dice scores were also the same. The major difference between the deformation modes are to be found where the warp displacement parameters are close to the limit $1/\pi$. The sites in the Cootes warp with highest curvature are closer to singular than the same sites in the warp based on our parameterization.

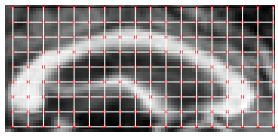


(a) (b)

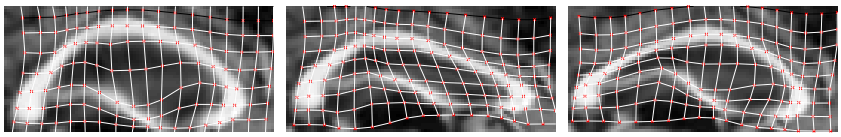
Figure 8.4: Plot relating described variance with number of modes included in the model. a: the presented method. b: method introduced by Cootes et al. [16].



(a) 1st mode, -3 std. dev. (b) 2nd mode, -3 std. dev. (c) 3rd mode, -3 std. dev.

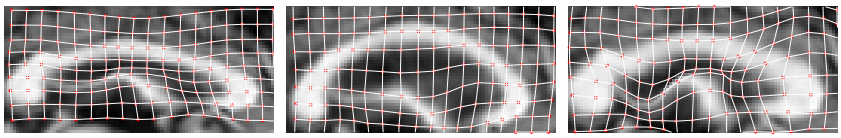


(d) Reference

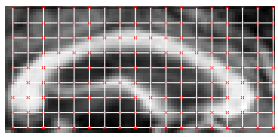


(e) 1 st mode, $+3$ std. dev. (f) 2nd mode, $+3$ std. dev. (g) 3rd mode, $+3$ std. dev.

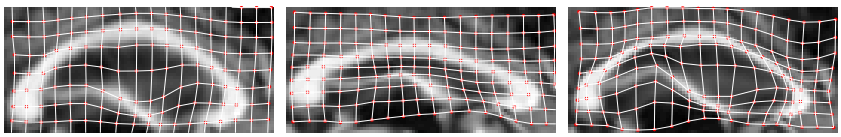
Figure 8.5: First three modes of the corpus callosum deformation model estimated with the current method, shown as the reference warped $\pm$ 3 std. deviations.



(a) 1st mode, -3 std. dev. (b) 2nd mode, -3 std. dev. (c) 3rd mode, -3 std. dev.



(d) Reference



(e) 1 st mode, $+3$ std. dev. (f) 2nd mode, $+3$ std. dev. (g) 3rd mode, $+3$ std. dev.

Figure 8.6: First three modes of the corpus callosum deformation model estimated with a constrained version of Cootes' method, shown as the reference warped $\pm$ 3 std. deviations

8.5 Discussion

We have shown how a parametric function can be defined on the unbounded linear space $\mathbb{R}^M$ and still produce diffeomorphic warps. When this is accomplished by first mapping $\mathbb{R}^M$ into an open bounded subset of $\mathbb{R}^M$, which inevitably leads to an asymptotic behavior at the closure of the bounded set. In our implemented example the parameters of the model by Cootes et al. asymptotically approach $\frac{1}{\pi}$ where singularities in the warp *may* occur. We believe that our distance measure is very reasonable when we are indeed approaching a singularity, as a small change in the displacement parameters of the warp will cause a huge impact on curvature of the warp function. In Fig. 8.7, where -6 std. deviations of the first mode is shown. We see that a singularity start to form in the contracting area but this is highly unlikely as predicted by our model and metric.

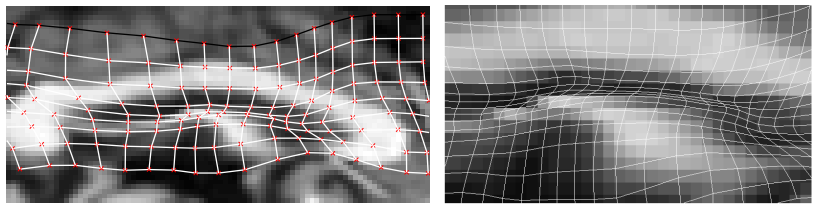


Figure 8.7: -6 Std. deviations of the first mode, normal view and a zoomed view on the beginning singularity.

With the choice of $\tanh$ function, the asymptotic behavior is assumed to be exponential, which may not always be the case. There are obviously an infinite variety of monotonic bounded functions, e.g. arcus tangent, and we will be investigating the choice of function in more detail.

A problem, we believe, that may occur with the proposed method is that we cannot be sure that the threshold does actually mark a singularity. A simple translation would for instance be asymptotic as well, which is why initial rigid alignment is very important indeed. Currently we investigate more involved parameter restrictions than the simple threshold to circumvent this possible problem.

Our validation on corpus callosum data showed that we were able to learn the important modes of variation, similar to previous obtained results, while the relatively high Dice coefficient illustrated that our warp representation was able to capture the large variations in the data set. We believe it is an advantage that all configurations in our parameter space are valid diffeomorphism, such that all gradients and derivatives during the optimization are well defined. Also we find it an advantage for tracking etc.

that the the deformation as a function of the deformation model parameters is smooth, when using the presented method.

8.6 Conclusions

This paper proposed a new warp representation which allows statistical analysis on an unrestricted linear parameter space, where all derivatives are defined. Furthermore, we have shown that L_2-norm the parameter space introduces a reasonable metric in the actual space of modelled diffeomorphisms, and that our results compare well to those obtained using Cootes' deformation model.

Acknowledgements

The authors extend their gratitude to the LADIS work group for supplying the corpus callosum data. In particular we acknowledge the annotation effort of Charlotte Ryberg and Egill Rostrup from the Danish Research Center for Magnetic Resonance, Copenhagen University Hospital, Hvidovre, Denmark. We also wish to thank Vagn L. Hansen and Niels V. Christensen for fruitful discussion during this work.

8.7 APPENDIX

8.7.1 Warp inversion

THEOREM 8.1 *Consider the function* $\boldsymbol{\varphi} : \mathbb{R}^N \times \mathbb{R}^M \mapsto \mathbb{R}^N$ *of type* $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}) = \boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x}, \boldsymbol{t})$ *and let* $\boldsymbol{\varphi_t}(\boldsymbol{x}) = \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t})$ *be a* $\mathcal{C}^1$*-diffeomorphism. If* $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{0}) = \boldsymbol{0}$ *and* $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{t}) = -\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{t})$*,* $\boldsymbol{\varphi}(\boldsymbol{x}, -\boldsymbol{t})$ *converges with second-order to* $\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, \boldsymbol{t})$*.*

PROOF.

$$\begin{aligned} & |\xi_i(h\boldsymbol{t})| = |\varphi_i(\boldsymbol{\varphi}(\boldsymbol{x}, h\boldsymbol{t}), -h\boldsymbol{t}) - x_i| \\ = \; & |x_i + u_i(\boldsymbol{x}, h\boldsymbol{t}) - u_i(\boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x}, h\boldsymbol{t}), h\boldsymbol{t}) - x_i| \\ < \; & |u_i(\boldsymbol{x}, h\boldsymbol{t}) - u_i(\boldsymbol{x}, h\boldsymbol{t}) + \frac{\partial u_i}{\partial \boldsymbol{x}}(\boldsymbol{x}, h\boldsymbol{t})\boldsymbol{u}(\boldsymbol{x}, h\boldsymbol{t})| \\ < \; & |h\boldsymbol{t}^\top \frac{\partial^2 u_i}{\partial \boldsymbol{x} \partial h\boldsymbol{t}}(\boldsymbol{x}, \boldsymbol{0}) \frac{\partial \boldsymbol{u}}{\partial h\boldsymbol{t}}(\boldsymbol{x}, \boldsymbol{0}) h\boldsymbol{t}| \\ < \; & |c| \cdot |h^2| \end{aligned} \tag{8.19}$$

8.7.2 Derivation of update function

In general, it is unlikely that $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}) \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{x}, \Delta\boldsymbol{t})$ can be parameterized with $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}')$, and thus it has to be approximated.
In Appendix 8.7.1, it was shown that $\boldsymbol{\varphi}(\boldsymbol{x}, -\boldsymbol{t})$ is a first-order approximation to $\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, \boldsymbol{t})$ as the error converges with second-order to zero. The composition in Eq. 8.15 is approximated with the parameters $\boldsymbol{t}'$ which minimizes the SSD between the true compositional warp and the warp $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}')$

$$\sum_x \Delta\boldsymbol{\varphi}(\boldsymbol{x})^\top \Delta\boldsymbol{\varphi}(\boldsymbol{x}) \tag{8.20}$$

where

$$\begin{aligned} \Delta\boldsymbol{\varphi}(\boldsymbol{x}) &= \boldsymbol{\varphi}(\boldsymbol{\varphi}(\boldsymbol{x}, \Delta\boldsymbol{t}), \boldsymbol{t}) - \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{t}') \\ &= \boldsymbol{a}(\boldsymbol{x})^\top (\boldsymbol{g}_\tau(\Delta\boldsymbol{t}) - \boldsymbol{g}_\tau(\boldsymbol{t}')) \\ &\quad + \boldsymbol{a}(\boldsymbol{\varphi}(\boldsymbol{x}, \Delta\boldsymbol{t}))\boldsymbol{g}_\tau(\boldsymbol{t}). \end{aligned} \tag{8.21}$$

If

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a}(\boldsymbol{x_1})^\top \\ \vdots \\ \boldsymbol{a}(\boldsymbol{x_n})^\top \end{bmatrix}, \text{and } \boldsymbol{A}_\varphi = \begin{bmatrix} \boldsymbol{a}(\boldsymbol{\varphi}(\boldsymbol{x}_1, \Delta\boldsymbol{t})^\top \\ \vdots \\ \boldsymbol{a}(\boldsymbol{\varphi}(\boldsymbol{x}_n, \Delta\boldsymbol{t})^\top \end{bmatrix}$$

the updated warp parameters $\boldsymbol{t}'$ can be found by solving the system

$$\boldsymbol{0} = \boldsymbol{A}(\boldsymbol{g}_\tau(\Delta\boldsymbol{t}_i) - \boldsymbol{g}_\tau(\boldsymbol{t}_i')) + \boldsymbol{A}_\varphi \boldsymbol{g}_\tau(\boldsymbol{t}_i). \tag{8.22}$$

The least square solution to the system is

$$\boldsymbol{t}_i' = \boldsymbol{g}_\tau^{-1} \left(\boldsymbol{A}^\dagger \boldsymbol{A}_\varphi \boldsymbol{g}_\tau(\boldsymbol{t}_i) + \boldsymbol{g}_\tau(\Delta\boldsymbol{t}_i) \right) \tag{8.23}$$

where $\boldsymbol{A}^{\dagger} = \left[\boldsymbol{A}^{\top}\boldsymbol{A}\right]^{-1}\boldsymbol{A}^{\top}$.

As A_{φ} has to be evaluated on warped points it is relatively computational expensive to evaluate. Thus, we perform a first-order Taylor expansion on A_{φ} and arrive at

$$\boldsymbol{t}_i' = \boldsymbol{k}^{-1}\left(\boldsymbol{A}^{\dagger}\boldsymbol{A}_{J_i}\boldsymbol{A}\boldsymbol{g}_{\tau}(\Delta\boldsymbol{t}_i) + \boldsymbol{g}_{\tau}(\Delta\boldsymbol{t}_i) + \boldsymbol{g}_{\tau}(\boldsymbol{t}_i)\right), \tag{8.24}$$

where

$$A_{J_i} = \boldsymbol{I} + \operatorname{diag}(\frac{\partial\boldsymbol{a}(\boldsymbol{x}_j)}{\partial x_i}^{\top}\boldsymbol{g}_{\tau}(\boldsymbol{t}_i))_{j=1\ldots n} \tag{8.25}$$

CHAPTER 9

Curl-gradient Image Warping

Introducing Deformation Potentials for Medical Image registration using Helmholtz Decomposition

Michael Sass Hansen, Niels Vorgaard Christensen and Rasmus Larsen [1]

Abstract

Image registration is becoming an increasingly important tool in medical image analysis, and the need to understand deformations within and between subjects often requires analysis of obtained deformation fields.

The current paper presents a novel representation of the deformation field based on the Helmholtz decomposition of vector fields. The two decomposed potential fields form a curl free field and a divergence free field. The representation has already proven its worth in fluid modelling and electrostatics, and we show it also has desirable features in image registration and morphometry in particular. The potentials are shown to a offer decoupling of the two potential fields in both elastic and fluid image registration. For morphometry applications, we show that when decomposing the deformation field in symmetric and antisymmetric parts, the vector potential alone describes the vorticity, and the scalar gradient potential gives a first-order approximation to the determinant of the Jacobian.

We provide some insight into the behavior of curl and divergence representation of the warp field by constructed examples and by a demonstration on real medical image

[1] This paper was presented at the International Conference on Computer Vision Theory and Applications 2009 [42]

data. Our theoretical findings are readily observable in our empirical experiment, which further illustrates the benefit of the parametrization.

9.1 INTRODUCTION

Image registration is becoming an increasingly important tool in medical image analysis, and the need to understand deformations within and between subjects often requires analysis of obtained deformation fields [100, 21].

The image registration task has been approached in many different ways. It can be achieved by calculation of dense deformation fields using nonparametric methods as described by Modersitzki [73]. Vermuri *et al.* proposed a levelset representation of the deformations [107]. For more than a decade parametric representation of the deformation fields has also been very popular in the literature, where two prominent examples are the cosine kernels presented by Cootes *et al.* [16], and the Cubic B-spline kernels used by Rueckert *et al.* [85]. The later introduces a certain smoothing by the finite size of the parametric kernel functions, but often further *regularization* is introduced. Haber and Modersitzki introduced regularization terms to ensure displacement regularity [37]. Elastic registration is a popular form of regularization, originating from continuum mechanics as described by Christensen and Johnson [12] and by Kybic *et al.* [62].

In the subsequent morphometry it has been shown that the Jacobian of the deformation field is very important under a Gaussian random field assumption on the deformation field [58]. Chung *et al.* have investigated different measures of morphometry, and also introduced a strain-curl representation of the deformation field, which has interesting relations to other morphometry measures [13]. Hsiao *et al.* have shown that using a parameterizing of the deformation field by its divergence and curl makes it less prone to grid folding than B-spline representation, while it allows for an efficient and stable optimization [57]. Kohlberger *et al.* have used potential functions for motion estimation of fluids [60].

In Section 9.2 we propose a new parametric representation of the deformation field, which is based on the Helmholtz decomposition of vector fields. In the following we show how this representation can also be parameterized by smooth kernels, how it can be considered a natural representation for elastic image registration, we show how it can be given a strong interpretation in morphometry, and finally we point out how the numerical stability and smoothness obtained by Hsiao *et al.* [57] can also be reached by our representation.

We have demonstrated an implementation of the presented method on MRI Corpus Callosa images from the midsagittal plane in Section 9.3.

9.2 METHODS

We present the developed methodology by introducing the Helmholtz decomposition representation in Section 9.2.1, with some intuitive demonstration of the introduced potential functions. Subsequently we give theoretical motivations by formulating simple elasticity regularization in Section 9.2.2, morphometric interpretation in Section 9.2.3, and finally we demonstrate properties for optimization in Section 9.2.4.

9.2.1 Helmholtz Decomposition of Vector Fields - Introducing Two New Deformation Potentials

We define a warp function $\boldsymbol{\varphi}$ as the mapping between two 3-dimensional images, the image I and the reference R, $\boldsymbol{\varphi} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, which satisfies that the point $\boldsymbol{x}$ in the reference R corresponds to the point $\boldsymbol{\varphi}(\boldsymbol{x})$ in the image I. The important aspect of this definition is that that $\boldsymbol{\varphi}$ can be considered a vector field. For medical image analysis purposes we can furthermore assume that the vector field is smooth, as this will usually be our best assumption for the anatomical topology. These considerations also holds for the deformation field, $\boldsymbol{u}$, which we define as the difference from the identity warp, such that $\boldsymbol{\varphi} = \boldsymbol{x} + \boldsymbol{u}$.

We apply the Helmholtz decomposition to the deformation field, using the fact that a vector field, which is twice continuously differentiable and with rapid enough decay at infinity, can be split into a sum of the gradient of a scalar function and the curl of a vector function [35]

$$\boldsymbol{u}(\boldsymbol{x}) = \nabla V(\boldsymbol{x}) + \nabla \times \mathbf{A}(\boldsymbol{x}), \tag{9.1}$$

where $V : \mathbb{R}^3 \rightarrow \mathbb{R}$ and $\mathbf{A} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ are scalar and vector potentials functions respectively. Sections 9.2.2,9.2.3 and 9.2.4 all deal with specific properties of this representation by potentials. Because they are new in the field of medical image analysis, we start by exploring some of the immediate properties of these potentials. Recall that the deformation field is merely a sum of the two, then we shall explore the gradient potential V, and the curl potential $\boldsymbol{A}$ one at a time.

9.2.1.1 The gradient potential V

The gradient potential is roughly speaking governing local contraction or expansion, this is in particular true in the presence of sufficiently small deformations. The same potential is used in electrostatics to describe the electrical potential. In Figure 9.1 gradient potentials and their impact on the deformation field are illustrated. It can be

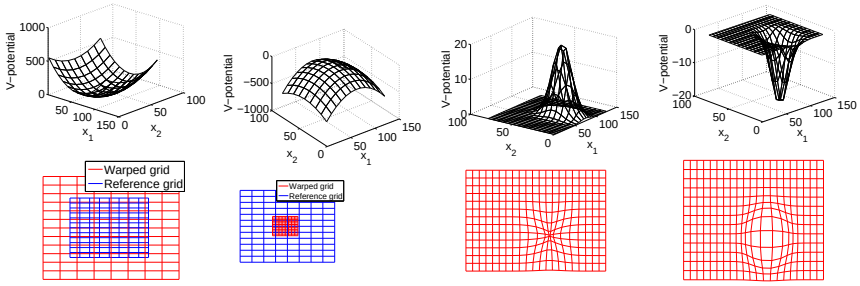


Figure 9.1: Gradient potentials, V, are illustrated in the upper row, and their impact on the deformation fields are shown below. The quadratic surfaces with constant curvature produce global scaling, and the potentials with local variations produce a local contraction or expansion.

observed how a positive versus a negative gradient results in expansion or contraction of the deformation fields. Two-dimensional functions are used for the purpose of illustration.

9.2.1.2 The curl potential $\boldsymbol{A}$

The curl potential field is the equivalent to the magnetic potential field in electrostatics. It is describing purely divergence-free deformations, which can be interpreted as vortices. In Figure 9.2 different curl potentials and their impact on the deformation field are illustrated. For the purpose of illustration only the z-component of the curl potential $\boldsymbol{A}$ is illustrated, and only the impact on the x, y directions of the deformation field are illustrated.

9.2.2 Elastic Registration

As previously mentioned, the elastic potential is often used for image registration, which is often based on a physical motivation in terms of an elastic tissue model [12, 62]. Regularization is usually formulated by a potential $\mathcal{S}$ and differential operators $\mathcal{B}$ [37]

$$\mathcal{S}[\boldsymbol{u}] = \int_{\Omega} \langle \mathcal{B}[\boldsymbol{u}], \mathcal{B}[\boldsymbol{u}] \rangle_{\mathbb{R}^d} \, d\boldsymbol{x} \; . \tag{9.2}$$

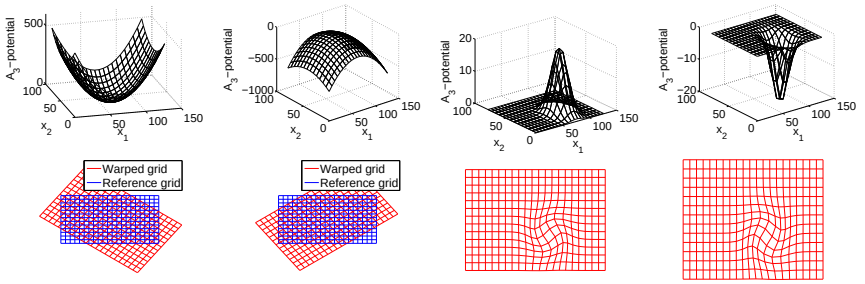


Figure 9.2: z-component, A_z of curl potential fields $\boldsymbol{A}$ are shown in the upper row, and the impact on the x, y coordinates of the deformation field are shown below. A quadratic potential with constant curvature produces a global rotation of the deformation field, and a local change to the potential results in a local vortex in the deformation field.

The corresponding Gâteaux derivative is then given by

$$d_{u;v}\mathcal{S}[\boldsymbol{u}] = \int_\Omega \langle \mathcal{A}[\boldsymbol{u}](\boldsymbol{x}), \boldsymbol{v}(\boldsymbol{x})\rangle_{\mathbb{R}^d}\, d\boldsymbol{x}\ , \tag{9.3}$$

where $\mathcal{A} = \mathcal{B}^*\mathcal{B}$. For the elastic potential this Navier-Lamé operator is given by

$$\mathcal{A} = \mathcal{B}^*\mathcal{B}[\boldsymbol{u}] = \mu\Delta\boldsymbol{u} + (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u}) \tag{9.4}$$

9.2.2.1 Elastic registration of the Helmholtz Decomposition

We examine how the elastic potential affects the Helmholtz decomposition of the warp field, and straightforward calculations give

$$\mathcal{P}[u] = \int_\Omega \mu \sum_{a,b=1}^{3} \epsilon_{ab}^2 + (\mu+\lambda)(\nabla\cdot u)^2 d\boldsymbol{x}\ , \tag{9.5}$$

where $\epsilon_{ij} = \delta u_i/\delta x_j + \delta u_j/\delta x_i$. Now $\boldsymbol{u} = \nabla V + \nabla\times\boldsymbol{A}$ and

$$\begin{aligned}\mathcal{A}[\boldsymbol{u}] &= \mu\Delta\boldsymbol{u} + (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u})\\ &= \mu\Delta(\nabla V + \nabla\times\boldsymbol{A})\\ &+(\lambda+\mu)\nabla(\nabla\cdot(\nabla V + \nabla\times\boldsymbol{A}))\\ &= (2\mu+\lambda)\nabla\Delta V + \mu\Delta\nabla\times A.\end{aligned} \tag{9.6}$$

This is a rather remarkable result, since we can now decouple the two potentials. Let $\lambda_1 = 2\mu+\lambda$ and $\lambda_2 = \mu$, then

$$\mathcal{A}[\boldsymbol{u}] = \lambda_1\nabla\Delta V + \lambda_2\Delta\nabla\times A \tag{9.7}$$

and replacing the Lamé constants μ and λ by λ_1 and λ_2, we have a clear notion of how to interpret the two regularization parameters. Not all values of λ_1 and λ_2 have meaning in a physical material-property sense (as goes for a negative μ as well), which should be considered. It can be argued, though, that the computational model should be extended to include these cases as well [73]. All derivations are also valid if we had differentiated in the time-domain, so our potential representation has the same advantages in fluid image registration.

9.2.3 Deformation-based Morphometry

Following Chung *et al.* we shall assume that the displacement field $\boldsymbol{u}$ is a smooth function in time, capturing the variation in shape over time [13]. In Section 9.3 we discuss how an artificial time can be introduced, even when we are doing inter-subject registration. Introducing deformation fields as a function of time, $\boldsymbol{u}(\boldsymbol{x}, t)$, the deformation field $\boldsymbol{u}$ at $\boldsymbol{x} + d\boldsymbol{x}$ can be written using a Taylor expansion

$$\boldsymbol{u}(\boldsymbol{x} + d\boldsymbol{x}, t) \approx \boldsymbol{u}(\boldsymbol{x}, t) + \boldsymbol{J_u} d\boldsymbol{x} \,, \tag{9.8}$$

where $\boldsymbol{J_u}$ denotes the Jacobian of the deformation field. It is usually assumed that the Jacobian contains all information relevant in morphometry, and still following Chung we shall look at a possible decomposition of it.

9.2.3.1 Vorticity and strain of the Jacobian

The Jacobian can be divided into symmetric and antisymmetric parts by the following decomposition

$$\frac{\partial u_j}{\partial x_i} \quad = \quad \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} - \frac{\partial u_i}{\partial x_j}\right) \quad + \quad \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right) \tag{9.9}$$

The first antisymmetric part is termed vorticity and the second part the strain. Using this (9.8) may be written as

$$\boldsymbol{u}(\boldsymbol{x} + d\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t) - \frac{1}{2}\nabla \times \boldsymbol{u}(\boldsymbol{x}, t) \times d\boldsymbol{x} + \boldsymbol{\epsilon}(\boldsymbol{x}, t) d\boldsymbol{x} \,, \tag{9.10}$$

where the strain matrix is given by $\boldsymbol{\epsilon} = (\epsilon_{ij}) = \frac{1}{2}\left(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}} + \left(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\right)^T\right)$. Observe that the vorticity depends on the $\boldsymbol{A}$ potential alone. Since $\nabla \times \nabla V(\boldsymbol{x}, t) = 0$ we get

$\nabla \times \boldsymbol{u}(\boldsymbol{x},t) = \nabla \times \nabla \times \boldsymbol{A}(\boldsymbol{x},t)$. The diagonal elements of the strain matrix $\boldsymbol{\epsilon}$ are in particular given by

$$\epsilon_{11} = \frac{1}{2}\left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_1}{\partial x_1}\right) = \frac{\partial\left(\frac{\partial A_2}{\partial x_3} - \frac{\partial A_3}{\partial x_2} + \frac{\partial V}{\partial x_1}\right)}{\partial x_1} \tag{9.11}$$

Taking the temporal derivative, the deformation velocity can be written as

$$\frac{\partial \boldsymbol{u}(\boldsymbol{x}+d\boldsymbol{x},t)}{\partial t} = \frac{\partial \boldsymbol{u}(\boldsymbol{x},t)}{\partial t} - \frac{1}{2}\frac{\partial}{\partial t}\nabla \times \boldsymbol{u}(\boldsymbol{x},t) \times d\boldsymbol{x} + \frac{\partial \boldsymbol{\epsilon}(\boldsymbol{x},t)d\boldsymbol{x}}{\partial t} \tag{9.12}$$

and in [13] it is shown that the first order approximation to the Jacobian determinant is the sum of the diagonal elements of the strain matrix $\boldsymbol{\epsilon}$;

$$\frac{\partial |\boldsymbol{J_u}|}{\partial t} \approx \frac{\partial \epsilon_{11}}{\partial t} + \frac{\partial \epsilon_{22}}{\partial t} + \frac{\partial \epsilon_{33}}{\partial t} = \frac{\partial}{\partial t}\left(\frac{\partial^2 V}{\partial x_1^2} + \frac{\partial^2 V}{\partial x_2^2} + \frac{\partial^2 V}{\partial x_3^2}\right). \tag{9.13}$$

This is seen to depend on the gradient potential alone, which can be understood, when we consider that this approximation of the Jacobian determinant is the divergence of the deformation field, $\nabla \cdot \boldsymbol{u}$. In summary we notice that our introduced representation gives several simplifications in relating our parameters to the morphological changes.

9.2.4 On Stability and Optimization

Hsiao *et al.* were using the curl and the divergence of the deformation field as parameters for image registration. Their experimental results showed this gave better stability in terms of avoiding grid folding than using a uniform B-spline parameterization [57]. In the current setting these quantities are given by $\nabla \times \boldsymbol{u} = \nabla \times \nabla \times \boldsymbol{A}$ and $\nabla \cdot \boldsymbol{u} = \nabla \cdot \nabla V = \Delta V$ respectively. The div-curl solver presented could be applied for our parameterizations as well, disregarding that we are presenting a different regularization term. However, for the morphometry test in Section 9.3 we have applied a parameterized variational approach described in Appendix 9.4, which demonstrates our method with elastic regularization. We believe that the method presented in the current work has a number of advantages. They have to make use of inversions of discretized operators to reconstruct the actual deformation field in the optimization step, which gives them a registration less prone to folding. We can simply use the exact differential operators on our potentials in order to arrive at actual deformations in our formulation, and use the regularization to enforce smoothness and invertibility.

9.3 RESULTS

To demonstrate that the described parameterization can also be practically implemented, we have implemented a cubic B-spline parameterization of the two potential fields. The implementation is described in more detail in Appendix 9.4, and in the next section we show results, and hope to add more intuition for the presented approach through visualization of the potentials on 2D spaces.

9.3.1 Morphometry on Corpus Callosum

The corpus callosum has been the subject of much analysis in the field of medical imaging [21, 85]. This is probably because of its relatively simple shape, and the good contrast in MRI. We have also chosen corpus callosum MR images sampled in the mid-sagittal plane to demonstrate the presented method. The data set used for the tests is a subset consisting of 62 MR images from the LADIS (Leukoaraiosis and DISability) study [78] - a pan-European study involving 12 hospitals and more than 700 patients. In the optimization we use an artificial time t, when registering one image to another [73]. Since the quantitative analysis is not the major objective of the current presentation, we put emphasis on illustrating properties of our potentials. In Figure 9.3 the image registration result of one corpus callosum to another is shown. We analyze the

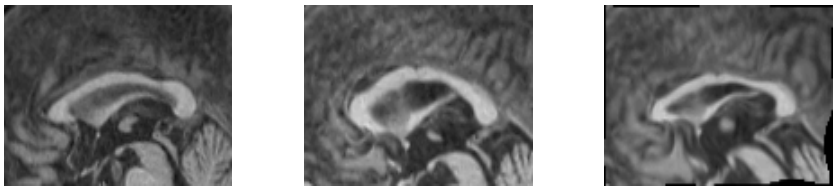
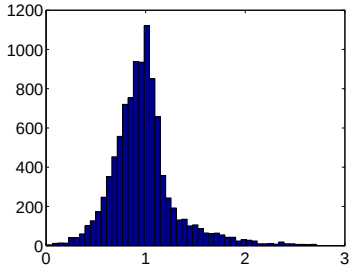


Figure 9.3: Registration by the proposed parameterization. The image in the middle is registered to the reference (left) and the result is shown to the right.

determinant of the deformation field to identify which areas are mostly deformed by the registration process. In Figure 9.4 the distribution of the determinant and the areas with significantly different values are illustrated. It is seen that the expansion (in this case) is most outspoken in 3 regions of the corpus callosum, which is not so surprising when we investigate the reference, and target image in Figure 9.3.

The potentials parameterizing the image registration are shown in Figure 9.5. It can be seen that the V-field is describing expansion and contractions, and several of the areas with interesting Jacobian determinant in Figure 9.4 are also seen to represent a



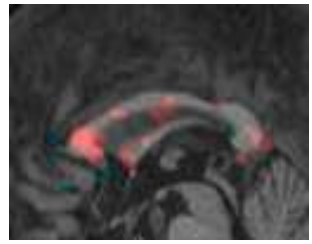


Figure 9.4: Above: The distribution of the determinant of the Jacobian. Below: Region of interest, where the deformation measured by the determinant, is outside a range of 2 std. dev..

rather strong contraction from the V-potentials. The $\boldsymbol{A}$-field is describing rotation - or vortices in the deformation field.

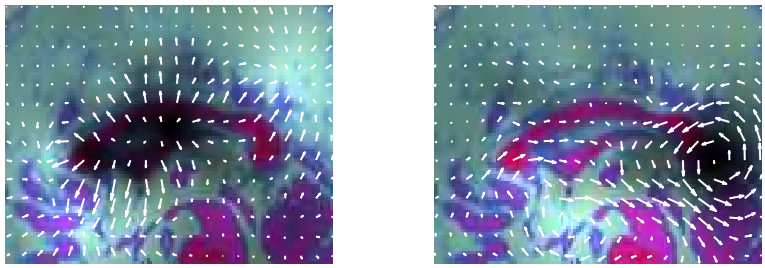


Figure 9.5: Potential functions shown with images and deformation fields using the HSV colormap. The potential is the value and the image is the hue. Left: V-potential along with (normalized) deformations this potential causes. Right: $\boldsymbol{A}$-potential, and the curl deformations this potential causes.

9.4 Discussion and conclusion

In this paper we introduce the theory of a new parameterization of 3D deformation fields to the field medical image analysis, using potentials and Helmholtz decomposition. Similar methods have already proven valuable in electrostatics and fluid flow estimation [35, 60]. We show the representation can be considered a natural parameterization for both elastic and fluid image registration due to the decoupling of the param-

eters. For morphometry we have demonstrated that one of the two potentials directly gives us the vorticity of the deformation field. The determinant gradient field is shown to be the first-order small-deformation approximation to the determinant of the Jacobian matrix - probably the most accepted morphometry measure used. Contemporary methods for optimization can supposedly be adapted to the parameterization [57, 60] and we have outlined our implementation based on finite differences, in Appendix 9.4.

The major contribution of the paper is primarily a theoretical one, but we have for demonstration purposes included 2D examples illustrating the relation between the potentials and the observed deformation fields. It shown that we can get sensible results, where most of the theoretical observations are readily recognizable from our empirical experiments, and we anticipate many applications in the field of morphometry. For future work we plan to design quantitative tests on different medical data sets, to add further empirical validation to the theoretic results demonstrated in the current paper, and to document the impact on achieved solutions.

APPENDIX

In this section we give an overview of implementation details that are not of great importance to the theoretical contributions of this paper. In Section 9.4 we introduce the uniform cubic B-spline that are used in our implementation. In Section 9.4 we show some details on their regularization, and in Section 9.2.4 we give some details on how the evaluations can be sped up.

B-spline representation of fields

We represented the potential fields by cubic B-splines, following [85]. So in summary the two potential fields are represented as

$$V = \sum_{k=0}^{3}\sum_{l=0}^{3}\sum_{m=0}^{3} B_k(r)B_l(s)B_m(t)v_{k,l,m} \tag{9.14}$$

$$\boldsymbol{A} = \sum_{k=0}^{3}\sum_{l=0}^{3}\sum_{m=0}^{3} B_k(r)B_l(s)B_m(t)\boldsymbol{a}_{k,l,m}\ . \tag{9.15}$$

Regularization of the B-splines

Applying the elastic constraints (9.6) to the B-spline fields, we get for the scalar potential

$$\nabla\Delta V = \nabla \sum_{k,l,m=0}^{3} (B_k''(r)B_l(s)B_m(t) + B_k(r)B_l''(s)B_m(t) + B_k(r)B_l(s)B_m''(t))v_{k,l,m} \tag{9.16}$$

and for the vector potential we get

$$\Delta\nabla \times \boldsymbol{A} = \Delta\nabla \times \sum_{k=0}^{3}\sum_{l=0}^{3}\sum_{m=0}^{3} B_k(r)B_l(s)B_m(t)\boldsymbol{a}_{k,l,m} \tag{9.17}$$

The regularization of the control points is now determined by

$$\frac{\delta\mathcal{S}[\boldsymbol{u}]}{\delta v_{lkm}} = d_{\boldsymbol{u},\frac{\delta\boldsymbol{u}}{\delta v_{klm}}} S[\boldsymbol{u}] = \int_\Omega \left\langle \mathcal{A}[\boldsymbol{u}](\boldsymbol{x}), \frac{\delta\boldsymbol{u}}{\delta v_{klm}} \right\rangle d\boldsymbol{x} \tag{9.18}$$

Optimization

We note that in our parameterized setting, the $\mathcal{A}$-operator can be written as linear combination of the parameters $\mathcal{A} = K_{\mathcal{A}}\boldsymbol{p}$, as can the warp field $\boldsymbol{u} = K_{\boldsymbol{u}}\boldsymbol{p}$. Using this (9.18) can be rewritten as

$$\frac{\delta\mathcal{S}[\boldsymbol{u}]}{\delta v_{lkm}} = d_{\boldsymbol{u},\frac{\delta\boldsymbol{u}}{\delta v_{klm}}} S[\boldsymbol{u}] = \int_\Omega (K_{\mathcal{A}}\boldsymbol{p})^T K_{\boldsymbol{u},klm} d\boldsymbol{x} = \boldsymbol{p}^T \int_\Omega K_{\mathcal{A}}^T K_{\boldsymbol{u},klm} d\boldsymbol{x} \tag{9.19}$$

it is seen that the Gateaux derivative is indeed linear in the parameters, and the integral needs only be evaluated once. The distance measure between reference and image can for instance be the L_2-norm

$$\mathcal{D}[R, T; \boldsymbol{u}] = \int_\Omega \left[T(x + \boldsymbol{u}) - R(x)\right]^2 dx \tag{9.20}$$

with the Gateaux derivative

$$d_{\boldsymbol{u};\boldsymbol{v}}\mathcal{D}[R,T;\boldsymbol{u}] = \int_{\Omega} \langle \boldsymbol{f}(x,\boldsymbol{u}(x)),\boldsymbol{v}\rangle_{R^d}\, dx$$

$$\begin{aligned}\text{where } \boldsymbol{f}(x,\boldsymbol{u}(x)) \\ =&\nabla T(x+\boldsymbol{u}(\boldsymbol{x}))(T(\boldsymbol{x}+\boldsymbol{u}(\boldsymbol{x}))-R(\boldsymbol{x}))\end{aligned} \tag{9.21}$$

And the same kernel substitutions can be made as for the regularization. This facilitates a quick estimation of the deformation field. A further speed up is gained by implementing a multi-grid cubic B-spline approach has been used, which also helps avoid local minima. In our presented results we used control point distances of 5, 10 and 20 pixels, respectively.

Part III

Multivariate splines

CHAPTER 10

Introduction to splines and adaptive parametrization

The word *spline* originally stems from the wooden rods shipbuilders were using as drawing devices for drawing bent curves [20]. They were chosen because they were producing smooth curves by minimizing the bending energy, and this property has also spiked the interest in splines as a mathematical tool. B-splines appeared in connection with Steklov means, which was observed by L. Maurer in 1896 [71]. They were studied from a statistical point of view by Sommerfeld [98] and Pólya [80]. In 1966 Curry and Schoenberg published a more contemporary formulation of the B-spline theory [18].

The extension of the curve splines to surfaces and more dimensions was initially approached by de P. de Casteljau and P. Bézier, published by Bézier in 1966, who introduced the Bézier patches [26, 10]. Michelli laid some of the foundation for multivariate splines in 1980 [72].

The work presented here focuses on multivariate splines, and their application in image analysis, and in particular image registration. Splines may be considered as sets of functions, which are each a compactly supported collection of piecewise polynomials. In the univariate case the cubic B-splines have been shown to minimize an approximation to the bending energy. Multivariate splines (including the univariate splines) are the Peano kernel in the integral representation of divided differences [7, 74]. That is, they define the local interpolation function, which interpolates the derivative of a func-

tion given the same result as the divided difference approximation to true derivatives. Multivariate splines are defined using knots, which are in general position, meaning that the control points are not constrained to follow any particular geometry. Splines are used for function approximation in many applications [26], and here some preliminary concerns on splines in image registration will be discussed.

10.1 Univariate linear splines

Some properties of the flexible representation can be illuminated by considering a simple 1 dimensional example of using splines. The one-dimensional linear B-spline may be defined by

$$b_{j,1}(t) = \begin{cases} \frac{t-t_j}{t_{j+1}-t_j} & \text{if} \quad t_j \leq t \leq t_{j+1} \\ \frac{t_{j+2}-t}{t_{j+2}-t_{j+1}} & \text{if} \quad t_{j+1} \leq t \leq t_{j+2} \\ 0 & \text{otherwise} \ , \end{cases}$$

where the knots $\{t_1, \ldots, t_{n+2}\}$ are a non-decreasing sequence of real number, j is an index specifying the spline, $j \in \{0, \ldots, n\}$. An example of 3 linear B-splines is shown in Figure 10.1.

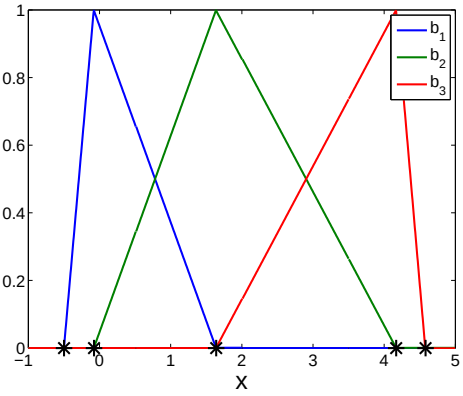


Figure 10.1: 3 linear B-spline basis functions. The knots are illustrated by black '*'.

If c_j is the control point value associated with the jth B-spline, then a function may be approximated by the linear combination of the B-spline basis functions

$$p(t) = \sum_{j=1}^{n} c_j B_{j,1}(t) \ .$$

A linear combination of B-splines is illustrated in Figure 10.2.

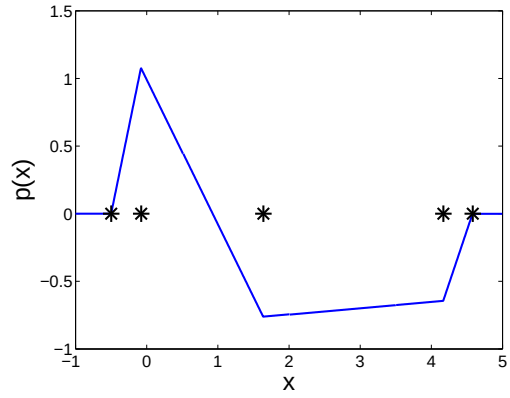


Figure 10.2: Linear combination of three B-spline basis functions illustrated in Figure 10.1. The weights c_1,c_2 and c_3 are $1.078, -0.763$ and -0.644 respectively.

10.2 Registration of functions

Let the two functions $f_i : \mathbb{R} \to \mathbb{R}$ $(i \in \{1, 2\})$, be such that one is a locally stretched and compressed version of the other, and let $\varphi : \mathbb{R} \to \mathbb{R}$ be the function describing the correspondences, ie. $f_2 \circ \varphi(x) = f_1(x),\ x \in \Omega$, and let u be the displacements $u = \varphi - x$. The functions f_1 and f_2 could for instance be two time series. It is instructive to examine the process of estimating this function u. In Figures 10.3, 10.4 and 10.5, a set of these functions is illustrated.

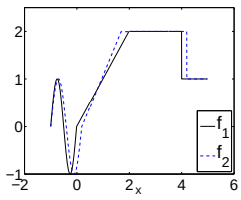


Figure 10.3: f_2 is a locally deformed version of f_1.

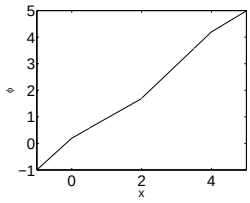


Figure 10.4: φ, mapping from f_2 to f_1.

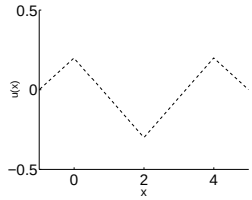
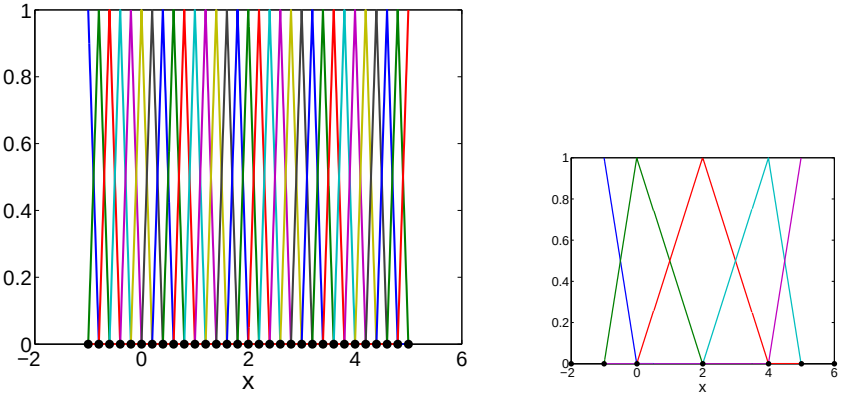


Figure 10.5: u measures the local deformation.

In order to estimate the function u a basis of B-splines is constructed, and the problem is formulated as the minimization of the squared residual between the composed $f_2 \circ \phi$ and f_1 with respect to the control points c_j. Two different linear B-spline bases, $\{b_{j,u}\}$

with uniform knots and $\{b_{j,n}\}$ with nonuniform knots, are analyzed. Both bases are shown in Figure 10.6.



(a) Uniform knots, $\{b_{j,u}\}$ (b) Adaptive knots, $\{b_{j,n}\}$

Figure 10.6: (a) shows basis functions based on uniformly sampled knots, and (b) shows basis functions of knots that are adapted to the application.

The minimization problem is posed as

$$\min_{\{c_j\}} \sum_x \left[f_1(x) - f_2\left(x + \sum_j c_{j,\cdot} \cdot b_{j,\cdot}(x) \right) \right]^2 + \alpha \|\boldsymbol{L}\boldsymbol{c}\|_2^2 \,, \tag{10.1}$$

where the first sum is taken over a set of x samples, $\boldsymbol{L}$ is a usual Tikhonov regularizer in the form of a matrix which is taken to be the identity matrix in the current example, $\boldsymbol{c}$ is a vector containing the elements $c_{j,\cdot}$ and α is the regularization parameter. This problem is nonlinear in nature due to the composition of f_2 with $\varphi = x + \sum_j c_{j,\cdot} \cdot b_{j,\cdot}(x)$, however it can be solved by gradient based methods and local linearization. For an analysis of the optimization problem (10.1) the difference is linearized in the variables $\{c_{j,\cdot}\}$ by forming a first-order Taylor expansion

$$f_2\left(x + \sum_j c_{j,\cdot} \cdot b_{j,\cdot}(x) \right) \approx$$
$$f_2\left(x + \sum_j c^0_{j,\cdot} \cdot b_{j,\cdot}(x) \right) + \left. \frac{\partial f_2}{\partial x} \right|_{x + \sum_j c^0_{j,\cdot} \cdot b_{j,\cdot}(x)} \sum_j \Delta c_{j,\cdot} \cdot b_{j,\cdot}(x) \,.$$

Let the sample points be numbered and ordered in a set $\{x_i\}$, and introduce the matrix $\boldsymbol{A}$ with elements $A_{ij} = \left. \frac{\partial f_2}{\partial x} \right|_{x_i + \sum_j c^0_{j,\cdot} \cdot b_{j,\cdot}(x)} \cdot b_{j,\cdot}(x_i)$. Furthermore, let the residual

error given the Taylor expansion point $\boldsymbol{c}_0$ be a vector $\boldsymbol{b}$ with elements $b_i = f_1 - f_2\left(x + \sum_j c^0_{j,\cdot} \cdot b_{j,\cdot}(x)\right)$ and the vector $\Delta\boldsymbol{c}$ have the elements Δc_i such that we have linearized $\boldsymbol{c}$ around $\boldsymbol{c}_0$, then the linearized problem, using the Taylor expansion, may be written as the following optimization problem

$$\|\boldsymbol{A}\Delta\boldsymbol{c} - \boldsymbol{b}\|_2^2 + \alpha\|\boldsymbol{L}(\boldsymbol{c}_0 + \Delta\boldsymbol{c})\|_2^2 \tag{10.2}$$

This approximated optimization problem will be discussed in more detail in Section 10.3.

10.3 The Picard condition

It seems worthwhile to consider which problems can actually be solved by a certain discretization, as well as the stability and the convergence properties of the discretization in question. P. Hansen introduced the discrete Picard condition for discrete problems, which loosely defined gives insight into wether a set of kernels contained in $\boldsymbol{A}$, defined in (10.2), can be expected to reproduce the vector $\boldsymbol{b}$ and can give some insight to the challenges of a parametrization of the type considered here [52]. The discrete Picard condition for discrete problems is introduced for the discretized Tikhonov-regularization problem

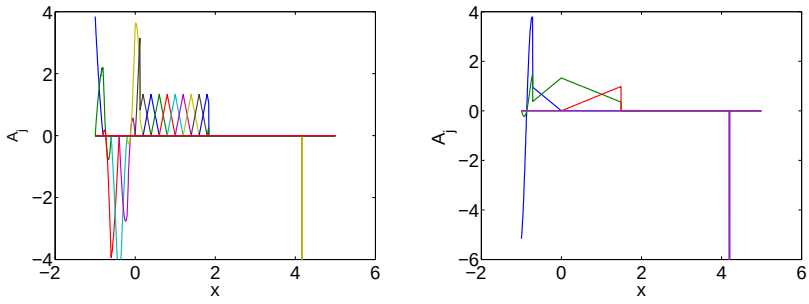
$$\boldsymbol{x}_\alpha = \min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \alpha\|\boldsymbol{L}\boldsymbol{x}\|_2^2 \tag{10.3}$$

which for the current analysis is equivalent to the optimization problem (10.2). Performing generalized singular value decomposition of the matrix set $(\boldsymbol{A}, \boldsymbol{L})$ gives the generalized singular values γ_i as well as the corresponding eigenvectors $\boldsymbol{u}_i$ [25].

THEOREM 10.1 (THE DISCRETE PICARD CONDITION (DPC).) *Let $\boldsymbol{b}$ be defined as in (10.3). Then $\boldsymbol{b}$ satisfies the DPC if, for all numerically nonzero generalized singular values γ_i, the corresponding Fourier coefficients $|\boldsymbol{u}_i^T\boldsymbol{b}|$ on the average decay faster to zero than the γ_i.*

Returning to the linearized optimization problem (10.2), this analysis may be applied to the problem during the optimization procedure. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm is applied for the optimization [9, 29, 33, 92].

When the optimization algorithm is started, all parameters $\boldsymbol{c}_0$ are initialized close to zero, and the plots of the matrix $\boldsymbol{A}$ can be seen in Figure 10.7. It is evident that for x in the range 2 to 4, there is no $\boldsymbol{A}$ response, since the gradient of f_2 is zero. It is seen from the uniform knot placement that a lot of redundant information is available, giving the same forces on most splines for x in the interval 0 to 2.



(a) Uniform knots, $\{b_{j,u}\}$ (b) Adaptive knots, $\{b_{j,n}\}$

Figure 10.7: (a) shows the matrix $\boldsymbol{A}$ based on uniformly sampled knots, and (b) shows $\boldsymbol{A}$ based on functions of knots that are adapted to the application.

The GSVD analysis of the linearization shows a division of the eigenvectors in higher and lower frequency components. In Figure 10.8 every second of the eigenvectors with nonzero generalized eigenvalues λ_i are shown, and this spectral representation of the matrix $\boldsymbol{A}$ illustrates how the parameters are able to reproduce the target function $\boldsymbol{b}$, which is illustrated in Figure 10.9.

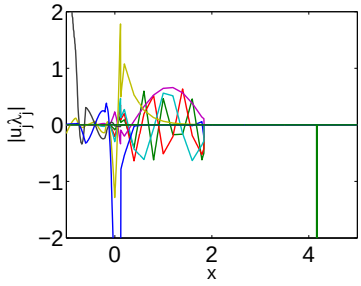


Figure 10.8: Illustration of the set of eigenvectors for the uniform B-splines.

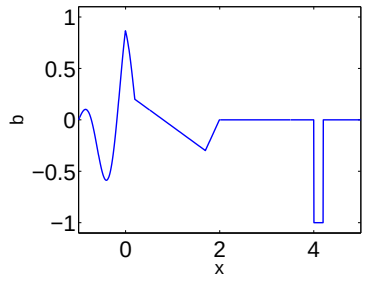
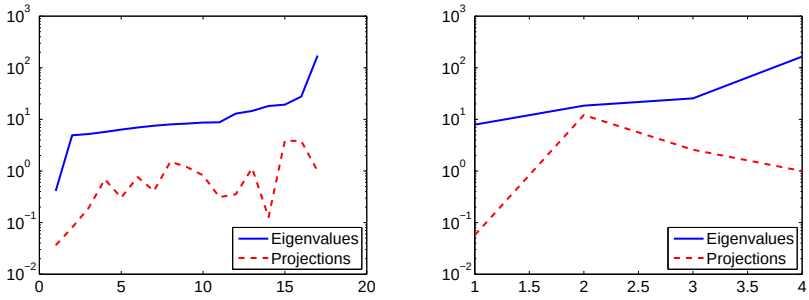


Figure 10.9: The residuals $\boldsymbol{b}$ when the optimization algorithm is initialized.

Generally, it appears quite plausible for the basis functions to be able to represent the residuals $\boldsymbol{b}$, and in Figure 10.10 the generalized eigenvalues are seen as well as the projection of $\boldsymbol{b}$ onto the eigenvectors.

For both the uniform and the nonuniform basis, the projections are seen to decay at least as rapidly as the eigenvalues them selves, from which it is concluded that the optimization problem is properly regularized. In fact, this only depends slightly on the regularization parameter α, so it appears that the implicit regularization imposed by the discretization and basis functions is already a good regularization for the stated



(a) Uniform knots, $\{b_{j,u}\}$ (b) Adaptive knots, $\{b_{j,n}\}$

Figure 10.10: (a) shows basis functions based on uniformly sampled knots, and (b) shows basis functions of knots that are adapted to the application.

problem. For the nonuniform knot positions it may be difficult to discuss the behavior of the eigenvalues versus the projections on the eigenvectors for the small number of basis functions. However, the tendency of faster decreasing projections $|u_j \boldsymbol{b}|$ from Figure 10.10 persists during the course of optimization, i.e. if the linearization is done at different steps of the BFGS optimization.

It is interesting to see the analysis performed at an example obtained by adding some noise to the original example. In Figure 10.11 functions f_1^* and f_2^*, which are generated from f_1 and f_2 by adding noise, are illustrated. In this more realistic example, the linearization looks somewhat different as seen in Figure 10.12, where every second column of the matrix $\boldsymbol{A}$ is illustrated (compare to Figure 10.7).

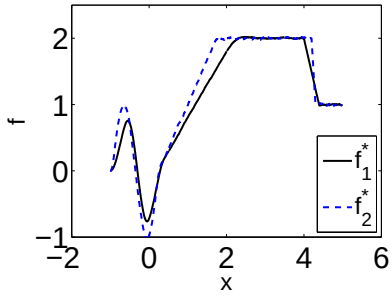


Figure 10.11: Functions f_1 and f_2 with added Gaussian noise and smoothed to form f_1^* and f_2^*.

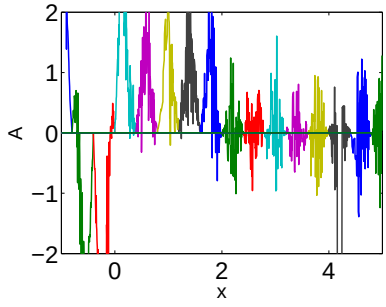
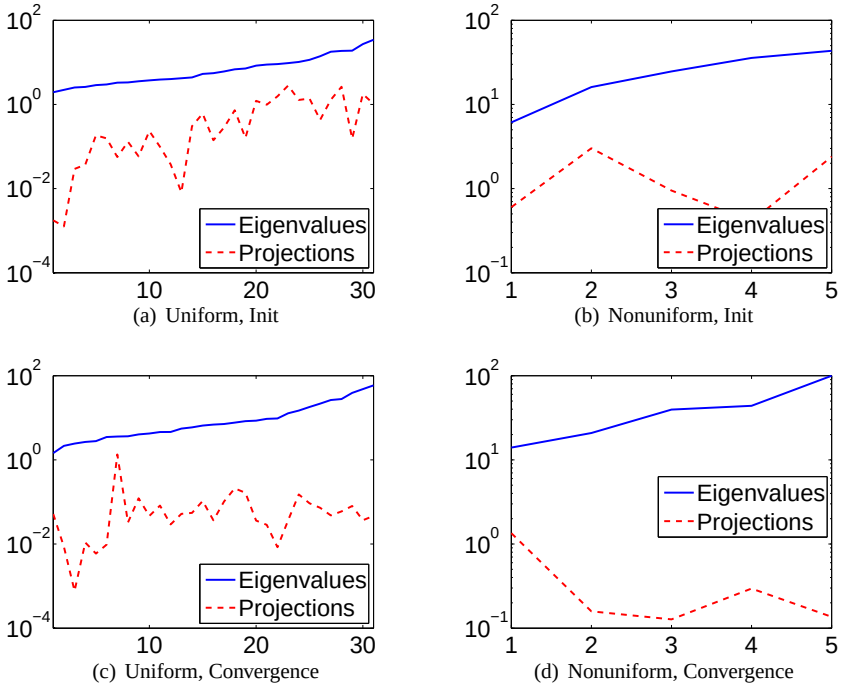


Figure 10.12: Illustration of the linearization matrix $\boldsymbol{A}$.

The added noise adds a response to all those basis functions that were having zero response in the noise free case because the gradient is not zero in this case. Examining the discrete Picard condition becomes interesting for these examples of functions. In Figure 10.13, the generalized eigenvalues and the projections on the eigenvectors can be visually examined, and it is evident that the Picard condition from Theorem 10.1 is satisfied for both bases initially, whereas close to the optimal point, the nonuniform sparse representation shows the opposite relationship between the two curves. The result that the linearization of the reduced model is not sufficiently regularized, hints at the fact that the high frequencies of the $\boldsymbol{A}$ basis are needed to do the optimization. For the less sparse example of uniform basis functions, the ability to describe these frequencies are better build into the model, and therefore no regularization is needed to make the problem numerically stable. It is hypothesized that the Picard condition will actually contribute information about the descriptive power of the basis and if it is sufficient for the problem, i.e. if refinement is needed.



(a) Uniform, Init (b) Nonuniform, Init

(c) Uniform, Convergence (d) Nonuniform, Convergence

Figure 10.13: Visual examination of the Picard condition for noisy functions, at the beginning and at the convergence of the optimization, for both the uniform and the nonuniform sets of knots.

10.4 Condition numbers and the stability of optimization

The condition number of a matrix is useful for examining the stability of matrix inversion and in particular the stability of solving linear systems with respect to small perturbations in the observations. The condition number of a matrix $\boldsymbol{M}$ is defined as

$$\kappa(\boldsymbol{M}) = \frac{\sigma_n}{\sigma_1} \tag{10.4}$$

where σ_n and σ_1 are the largest and smallest singular values of $\boldsymbol{M}$ respectively. A large difference in smallest and largest eigenvalues is thus contributing to a high condition number, which means the matrix is badly conditioned. The condition number sets a limit on the degree of impact a perturbation of the measured function $\boldsymbol{b}$ has on the solution $\boldsymbol{c}$, when solving the linear system $\boldsymbol{Ac} = \boldsymbol{b}$. In the noise free example, only 17 of the possible 31 generalized singular values are nonzero (Figure 10.10), which indicates a very badly conditioned system. Adding noise makes the system better conditioned, however noise is not adding information to the optimization process, only numerical stability, cf. Figure 10.13. Alternatively regularization may be introduced to add stability, but it also has an effect on the obtained solution [53]. Hansen showed that the tradeoff between regularization and function fitting for 'nice enough' functions has a identifiable optimum, which may be found using the L-curve. However, let two functions, defined on two intervals, each have optimum parameters in this way, then the union of the intervals and a function defined by a union of the functions may not have one optimal parameter, if the two parameters are not compatible. This combined function would not qualify as being 'nice enough', which adds motivation to introduce more local regularization, i.e. a different regularization on each interval. Returning to the optimization, if parameters are only well defined and robust to perturbations due to regularization, information from the *data* only propagates to the parameters through the regularization, and if the regularization has local (neighbor) impact, it requires more iterations to propagate the information to all parameters. In the previous noise-free example, both bases achieved the numerical optimum, but the badly conditioned uniform B-splines required 72 BFGS iterations, versus the 28 iterations of the nonuniform B-splines. Based on these considerations it may be hypothesized that a locally adjustable regularization, and a sparse representation of highly regularized areas can provide stable optimization, robust results, and efficient optimization. This can be considered an argument for parameterizing using nonuniform B-splines (or multivariate splines in higher dimension) since the restriction to the linear space spanned by the basis can be considered a local regularization, which can be made arbitrarily smooth by adjusting the order of the splines, and in highly regularized areas with large basis functions, the parametrization is sparse, which should yield faster convergence.

10.5 Conclusion

Based on the considerations in Section 10.4, it appears that multivariate B-splines may provide a good basis for parameterizing function approximation, and image registration in particular. Section 10.3 indicates that it may also be possible to assess if an approximating basis is suitable for a given problem, and these considerations are the motivations for the Chapters 13 and 12. Multivariate splines are introduced in Chapter 13, and it is demonstrated how they may be applied for *image registration*. And in Chapter 12 multivariate splines and *simplex splines* are discussed in more detail, developing methodology facilitating the goals motivated in the current chapter.

CHAPTER 11

Adaptive parametrization of B-splines for Image Registration

Michael Sass Hansen, Ben Glocker, Nassir Navab and Rasmus Larsen[1]

Abstract

We present an adaptive parametrization scheme for dynamic mesh refinement in the application of parametric image registration. The scheme is based on a refinement measure ensuring that the control points give an efficient representation of the warp fields, in terms of minimizing the registration cost function. In the current work we introduce multivariate B-splines as a novel alternative to the widely used tensor B-splines enabling us to make efficient use of the derived measure.

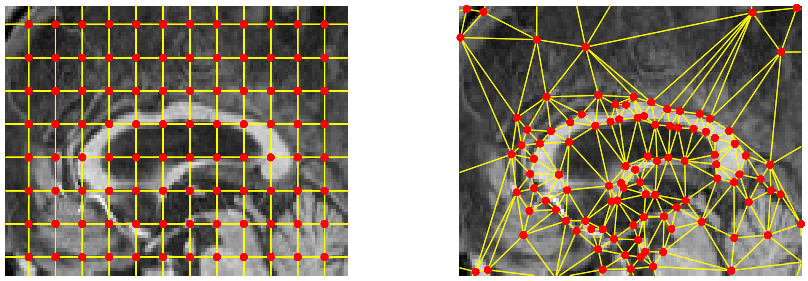
The multivariate B-splines of order n are C^{n-1} smooth and are based on Delaunay configurations of arbitrary 2D or 3D control point sets. Efficient algorithms for finding the configurations are presented, and B-splines are through their flexibility shown to feature several advantages over the tensor B-splines. In spite of efforts to make the tensor product B-splines more flexible, the knots are still bound to reside on a regular grid. In contrast, by efficient non-constrained placement of the knots, the multivariate B-splines are shown to give a good representation of inhomogeneous objects in natural settings.

[1]This paper was presented at International Conference for Computer Vision and Pattern Recognition [43]

The wide applicability of the method is illustrated through its application on medical data and for optical flow estimation.

11.1 Introduction

Image alignment is a challenging task due the inferior number of constraints compared to the number of unknowns. One approach to overcome this problem is to reduce dimensionality of the problem. Hence, parametric image registration has become quite popular and has been successfully applied to many applications. The two main advantages are (i) the reduced number of parameters which makes the problem tractable from an optimization point of view and (ii) often the transformation model inherently provides some smoothness properties on the warping field. In order to be able to capture the present deformations, hierarchical approaches have been considered where the number of parameters is successively increased during the optimization process. However, these update schemes often follow some heuristic approaches, for instance in mesh-based models by simply reducing the spacing between control points and thus do not incorporate any quantitative or qualitative measurement about the state of the registration. One can claim, that the refinement strategy has a significant influence on the solution and should be guided by some knowledge obtained directly from the images and/or the optimization process. Furthermore, one can claim that the transformation model should be flexible enough and dynamically adjustable in order to reflect such extracted knowledge during image alignment. Therefore, we propose a framework for parametric image registration which allows us on one hand to assess the quality of the current solution locally and on the other hand we can estimate the potential improvement by a local refinement of the parameter set. Since, one could expect that such an improvement will spatially vary over the image domain, there is a need for flexible transformation models.



(a) Regular B-spline grid (b) Multivariate B-spline knots

Figure 11.1: In (b) it is seen how the multivariate B-splines can adopt to the structure in the image.

B-splines are popular in numerous applications because of achievable smoothness properties and the local support. Tensor product B-splines were introduced for modeling

free form deformations (FFD) in the context of computer graphics [89]. The method was improved by using hierarchical B-splines [30, 113]. The same subdivision scheme was used in medical image registration [86]. However, the tensor product B-splines are defined uniformly on a grid, and in the search for a more local control of the representation detail, the notion of an *active set of knots* was introduced in [87]. Still, such an simulation approach consists of locally uniform grids which cannot represent the imaged objects in an appropriate way.

Cootes et al. created a minimum description length optimization scheme for the placement of control points of triangular piecewise affine interpolation basis functions [17]. However, the method is defined as a groupwise method, and it relies heavily on the calculation of the inverse deformation. Recently Chandrashekara et al. proposed a parameterization based on a subdivision scheme, also to obtain a limited number of parameters [83].

We instead propose to use the multivariate B-splines as presented by Neamtu, to obtain a flexible and smooth warp interpolation function with local support[75]. They are capable of representing fields more densely in some areas, while keeping the number of degrees of freedoms small for an efficient optimiation.

Dahmen et al. [19] introduced the so called DMS-splines or triangular B-splines based on the multivariate simplex splines and auxiliary knots. Franssen et al. [31] described a new method for efficient calculation of triangular B-splines using an evaluation graph. Recently, Wang et al. [111] introduced DMS-splines in computer vision for nonrigid registration with rigid parts that defined by manual landmarks. The most recent multivariate B-splines were introduced by Neamtu [75], and they rely heavily on the new concept of Delaunay configurations [76].

The remainder of this paper is organized as follows: first we will present the general framework for parametric image registration. Based on this, we derive our adaptive parametrization scheme. In Section 11.4 we present the concept behind multivariate B-splines and their use in our framework for image registration. Experimental results are shown in Section 11.5 while the last Section concludes our paper.

11.2 Parametric Image Registration

The image registration problem can be formulated as the minimization of the functional $\mathcal{J}$ given by

$$\mathcal{J}[I, R, \boldsymbol{\varphi}] = \mathcal{D}\left[I, R; \boldsymbol{\varphi}\right] + \mathcal{S}[\boldsymbol{\varphi}] \tag{11.1}$$

where I is an s-dimensional deformable image, R is the target/reference image, $\boldsymbol{\varphi}$ is the mapping from R into corresponding points in I. In the present work we are focusing on

parametric image registration, which means that the warps can formally be represented by

$$\boldsymbol{\varphi} : \mathcal{R}^M \rightarrow (f : \mathcal{R}^s \rightarrow \mathcal{R}^s) \tag{11.2}$$

which means that the parameters of $\boldsymbol{\varphi}$ determines what the warp will look like. We will think of $\boldsymbol{\varphi}$ as being represented as

$$\boldsymbol{\varphi} = \sum_i \boldsymbol{c}_i K_i(\boldsymbol{x}) \ , \tag{11.3}$$

where K_i are the basis functions, and the vectors $\boldsymbol{c}_i = [c_{i1}...c_{is}]^T$ are the parameters associated with the basis function.

$\mathcal{D}[I, R; \boldsymbol{\varphi}]$ is a similarity measure and $\mathcal{S}[\boldsymbol{\varphi}]$ is a regularization measure, and we will discuss these functionals in the context of parametric warp representation, in the following sections.

11.2.1 Similarity Measures

As a similarity measure the sum of squared distances (SSD) is used, but this could be exchanged with any other common similarity measures.

$$\mathcal{D}[I, R; \boldsymbol{\varphi}] = \int_\Omega \left[I \circ \boldsymbol{\varphi}(x) - R(x)\right]^2 dx \tag{11.4}$$

For the subsequent analysis we shall need some of the derivatives of $\mathcal{D}[I, R; \boldsymbol{\varphi}]$. The Gâteaux derivative of $\mathcal{D}$ is given by

$$d_{\boldsymbol{\varphi},\boldsymbol{v}}\mathcal{D}[I, R; \boldsymbol{\varphi}] = \int_\Omega \langle \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\varphi}), \boldsymbol{v} \rangle_{\mathcal{R}^s} \, d\boldsymbol{x} \tag{11.5}$$

where $\boldsymbol{f}$ can be perceived as registration forces in the image, and is given by

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\varphi}(\boldsymbol{x})) = (I \circ \boldsymbol{\varphi}(\boldsymbol{x}) - R(\boldsymbol{x})) \cdot \nabla I \circ \boldsymbol{\varphi}(\boldsymbol{x}) \ , \tag{11.6}$$

which is also denoted as the *driving force* of the registration process.

The variation $\boldsymbol{v}(\boldsymbol{x})$ is restricted to the subspace spanned by the basis functions. The derivative with respect to a warp parameter c_{ij} is given by

$$\frac{\partial \mathcal{D}}{\partial c_{ij}} = d_{\boldsymbol{\varphi}, \frac{\partial \boldsymbol{\varphi}}{\partial c_{ij}}} \mathcal{D}[I, R; \boldsymbol{\varphi}] = \int_\Omega \left\langle \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\varphi}), \frac{\partial \boldsymbol{\varphi}}{\partial c_{ij}} \right\rangle_{\mathcal{R}^s} d\boldsymbol{x}$$

11.2.2 Regularization

Often some prior knowledge is available about the presented image registration problem. This prior knowledge can generally be expressed as a differential regularizer $\mathcal{B}$, and some Sobolev norm.

$$\mathcal{S}[\boldsymbol{\varphi}] = \int_\Omega \langle \mathcal{B}[\boldsymbol{u}], \mathcal{B}[\boldsymbol{u}] \rangle_{\mathcal{R}^s} \, d\boldsymbol{x} \;, \tag{11.7}$$

where $\boldsymbol{\varphi} = \boldsymbol{Id} + \boldsymbol{u}$. This norm has a Gâteaux derivative given by

$$d_{u;v}\mathcal{S}[\boldsymbol{u}] = \int_\Omega \langle \mathcal{A}[\boldsymbol{u}](\boldsymbol{x}), \boldsymbol{v}(\boldsymbol{x}) \rangle_{\mathcal{R}^d} \, d\boldsymbol{x} \;, \tag{11.8}$$

where $\mathcal{A} = \mathcal{B}^*\mathcal{B}$. For the present work the commonly used elastic regularizer is chosen, which can be represented by

$$\mathcal{B}[\boldsymbol{u}] = \begin{bmatrix} \sqrt{\mu} & 0 \\ 0 & \sqrt{2\mu + \lambda} \end{bmatrix} \begin{bmatrix} \nabla \times \boldsymbol{u} \\ \nabla \cdot \boldsymbol{u} \end{bmatrix} \tag{11.9}$$

from which $\mathcal{A} = \mu \Delta \boldsymbol{u} + (\lambda + \mu) \nabla (\nabla \cdot \boldsymbol{u})$. We can again form the derivative with respect to the parameter

$$\frac{\partial \mathcal{S}[\boldsymbol{u}]}{\partial c_{ij}} = \int_\Omega \left\langle \mathcal{A}[\boldsymbol{u}], \frac{\partial \boldsymbol{u}}{\partial c_{ij}} \right\rangle d\boldsymbol{x} \tag{11.10}$$

11.3 Adaptive Parametrization

After minimizing (11.7) the following equation must hold for the reached optimum.

$$\frac{\partial \mathcal{J}[I, R; \boldsymbol{\varphi}]}{\partial c_{ij}} = \frac{\partial \mathcal{D}[I, R; \boldsymbol{\varphi}]}{\partial c_{ij}} + \frac{\partial \mathcal{S}[\boldsymbol{u}]}{\partial c_{ij}} = 0 \,. \tag{11.11}$$

This is the parameterized version of the variational optimum

$$\boldsymbol{f}(\boldsymbol{x}) + \mathcal{A}[\boldsymbol{u}] = 0 \;\;, \;\; \boldsymbol{x} \in \Omega \tag{11.12}$$

Now observe that (11.11) can be interpreted as an averaged projection of the variational optimum (11.12)

$$\frac{\partial \mathcal{D}}{\partial c_{ij}} + \frac{\partial \mathcal{S}[\boldsymbol{u}]}{\partial c_{ij}} = \int_\Omega \left\langle f(\boldsymbol{x}, \varphi) + \mathcal{A}[\boldsymbol{u}], \frac{\partial \boldsymbol{u}}{\partial c_{ij}} \right\rangle d\boldsymbol{x} = 0$$

where the projection is performed on to the support of the warp parameter. This projection, derived from (11.3), is given by

$$\frac{\partial \boldsymbol{u}}{\partial c_{ij}}(\boldsymbol{x}) = \boldsymbol{e}_j K_i(\boldsymbol{x}), \;\; \boldsymbol{x} \in \Omega \;\;,$$

where $\boldsymbol{e}_j$ denotes the unit vector along the jth dimension and K_i is the basis function associated with the parameter c_{ij}. In popular terms, each of our parameters is designed to achieve the variational optimum (11.12) in *average* only, and for this reason it seems obvious to pose the question; how well is the variational optimum achieved? We propose to measure this fitness of a basis function in terms of the improvement we could achieve by replacing the basis function with several more local basis functions. We will assume that the basis function can be refined into several similar basis functions, only with a smaller support, and the response of these local basis functions can be modelled by applying a Gaussian filter on the force residues $\left\langle f(\boldsymbol{x},\varphi) + \mathcal{A}[\boldsymbol{u}], \frac{\partial \boldsymbol{u}}{\partial c_{ij}} \right\rangle$. The filter response will model the changes that can be achieved with a basis function refinement, when we choose the kernel size to be close to that of the refined basis functions.

Let $\mathbf{1} = \boldsymbol{e}_1 + ... + \boldsymbol{e}_s$ then the above consideration lead us to define a refinement measure F by

$$F[B_I] = \int_\Omega \mathcal{F}_{\sigma_I} * \left(\langle f(\boldsymbol{x},\varphi) + \mathcal{A}[\boldsymbol{u}], \mathbf{1}B_I(\boldsymbol{x})\rangle\right)^2 d\boldsymbol{x}\,, \tag{11.13}$$

where $\mathcal{F}_{\sigma_I}*$ denotes convolution with a Gaussian of a kernel width σ_I which should be chosen in the order of $\sigma_I = [\text{vol}[B_I]]^{1/s}/4$, where vol$[B_I]$ is the volume of the convex hull of the basis function support. A perfect fit would mean that the only way (11.12) was not satisfied would be noise, and $F[B_I]$ would then be 0. However, if there is spatial coherency in the forces, $f(\boldsymbol{x},\varphi)+\mathcal{A}[\boldsymbol{u}]$, then $F[B_I]$ will give an output suggesting to do a refinement of the mesh. This criteria should guide the refinement.

In order to make efficient use of the proposed adaptation scheme, we need a set of basis functions with spatially varying local support.

11.4 Multivariate B-splines

The multivariate B-splines presented in this paper are using a basis of simplex splines. These splines are smooth functions with local support. Several ways exist for composing sets of simplex splines to form a partition of unity, the most recent one, and the one presented in the current work, being multivariate B-splines based on Delaunay configurations [75]. Simplex splines and Delaunay configurations are briefly discussed here. Throughout the dimension is still denoted by s and the degree of the splines by n.

11.4.1 Simplex Splines

Simplex splines are defined iteratively with the zeroth order spline defined on the simplex of $s+1$ knots, e.g. a triangle in 2D.

$$M(\boldsymbol{x}|\{\boldsymbol{x}_1,...,\boldsymbol{x}_{s+1}\}) = \begin{cases} 1/\text{vol}[\{\boldsymbol{x}_1,...,\boldsymbol{x}_{s+1}\}], & \boldsymbol{x} \in \text{int}[\{\boldsymbol{x}_1,...,\boldsymbol{x}_{s+1}\}] \\ 0 & \boldsymbol{x} \notin \text{int}[\{\boldsymbol{x}_1,...,\boldsymbol{x}_{s+1}\}] \end{cases}$$

where M denotes the spline value, $\boldsymbol{x}$ is a point we wish to evaluate, $\boldsymbol{x}_i$ are the knot points of the simplex spline, int refers to the convex hull of the set of points, and vol is the volume of the convex hull.

The recurrence relation of the higher order simplex splines is given by

$$M(\boldsymbol{x}|\boldsymbol{X}) = \sum_{\boldsymbol{x}_i \in \boldsymbol{X}} \lambda_i M(\boldsymbol{x}|\boldsymbol{X}\backslash\{x_i\}) \quad \text{, where}$$
$$\sum_{\boldsymbol{x}_i \in \boldsymbol{X}} \boldsymbol{\lambda}_i = 1, \quad \sum_{\boldsymbol{x}_i \in \boldsymbol{X}} \lambda_i \boldsymbol{x}_i = \boldsymbol{x} \; . \tag{11.14}$$

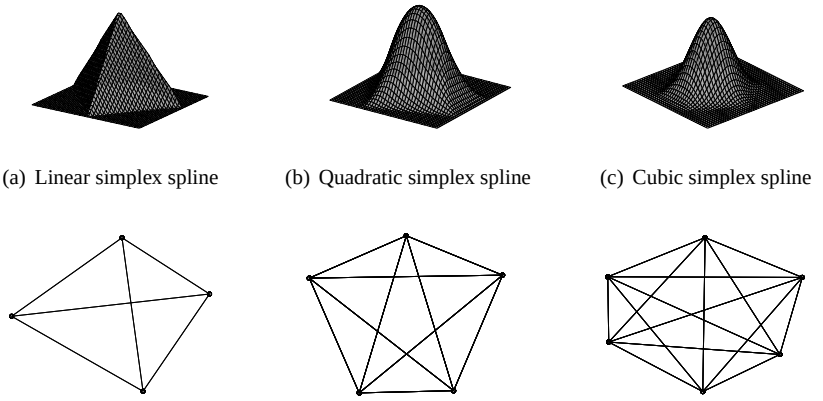
Here X is a set of $n+s+1$ knots (corresponding to a simplex spline M of order n, and $\boldsymbol{\lambda}$ is seen to be the barycentric coordinates of the points.

The simplex splines are $n-1$ smooth on the convex hull of the knots, when none three of the simplex splines are collinear [75]. Examples of 2D simplex splines of different orders are illustrated in Figure 11.2.

For the elastic regularization discussed in Section 11.6.2 the directional derivatives are needed, and the two first directional derivatives of the simplex splines are

$$d_{\boldsymbol{v}} M(\boldsymbol{x}|X) = n \sum_{i=0}^{s} \mu_i(\boldsymbol{v}) M(\boldsymbol{x}|X\backslash\{\boldsymbol{x}_i\})$$
$$d_{\boldsymbol{v}_2} d_{\boldsymbol{v}_1} M(\boldsymbol{x}|X) = n \sum_{i=0}^{s} \mu_i(\boldsymbol{v}_2) d_{\boldsymbol{x},\boldsymbol{v}_1} M(\boldsymbol{x}|X\backslash\{\boldsymbol{x}_i\}) \; ,$$
$$\text{where} \quad \sum_{i=0}^{s} \mu_i = 0, \quad \sum_{i=0}^{s} \mu_i \boldsymbol{x}_i = x$$

Complexity The complexity of the multivariate B-splines can be expressed in terms of the number of M_0 nodes visited, and this is n^{s+1} if a naive implementation is chosen. However, through fingerprinting visited nodes, this graph can be reduced considerably [31]. When calculating the interpolation values, the derivatives can be calculated simultaneously.



(a) Linear simplex spline (b) Quadratic simplex spline (c) Cubic simplex spline

(d) 4 knots for linear simplex (e) 5 knots for quadratic simplex (f) 6 knots for cubic simplex

Figure 11.2: Simplex splines of increasing order and smoothness.

11.4.2 Delaunay Configurations

For choosing appropriate sets of knots for simplex splines, the Delaunay configuration is needed. The Delaunay configuration is a generalization of the Delaunay triangulation, where the circumscribed sphere contains exactly n points. Denote a given Delaunay configuration of nth order by Δ_n. Let the set of all interior point sets, with n points in each be denoted by $\mathcal{I}$. Then a set of interior points $I \in \mathcal{I}$ is associated with a set of boundary point sets $\mathcal{B}(I) = \{B|(B, I) \in \Delta_n\}$. We now define the multivariate B-spline associated with n interior points I as [75]

$$B_I = \sum_{B \in \mathcal{B}(I)} \text{vol}[B] M(.|B \cup I) \tag{11.15}$$

This normalization ensures a partition of unity, i.e.

$$\sum_{I \in \mathcal{I}} B_I(x) = 1, \quad x \in \mathcal{R}^s \tag{11.16}$$

Using these multivariate B-splines as a basis for describing the deformation field, the field can be defined as

$$\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{x} + \sum_{I \in \mathcal{I}} \boldsymbol{c}_I B_I(\boldsymbol{x}), \quad \boldsymbol{c}_I, \boldsymbol{x} \in \mathcal{R}^s \tag{11.17}$$

Silveira et al. have shown a strategy for efficient computation of the Delaunay configurations [93].

11.4.3 Mesh Layout and Refinement

For deriving an initial guess for placement of the knots, it can be noted that (11.13) seems most likely to yield big values, in areas where the forces $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\varphi})$ attain big values. In terms, we expect the most changes to our deformation field to happen where the gradient is bigger, since the force and the gradient are proportional, (11.6). For the initial coarse distribution of knots, we propose to distribute them randomly, with a prior density based on the image gradient of the reference. Additionally, we add knots sequentially, according to the following update scheme

$$f(\boldsymbol{x}_{j+1}|R, \{\boldsymbol{x}_1, ..., \boldsymbol{x}_j\}) \approx f(\boldsymbol{x}_{j+1}|R) \prod_{i=1}^{j} f(\boldsymbol{x}_{j+1}|\boldsymbol{x}_i)$$
$$\propto \frac{(\nabla R)^T \nabla R}{\|\nabla R\|^2} \prod_{i=1}^{j} \left[1 - \exp\left\{-\frac{(\boldsymbol{x} - \boldsymbol{x}_i)^T(\boldsymbol{x} - \boldsymbol{x}_i)}{2\sigma}\right\}\right] ,$$

where ∇R is the gradient of the reference, and σ is a kernel diameter, where the kernel in effect limits the chance of a second point being place in the immediate vicinity of a knot. This function can be perceived as a prior probability for placing knots, and they can be placed either according to the maximum, or according to a random sampling. This did not seem to have big effect on results though, due to the subsequent mesh refinement.

We propose to make the mesh refinement based on subdivision of the grid of the knots, where the B-splines are expected to give most improvement in the cost function, when subdivided, according to the refinement measure derived in Section 11.3. To increase the resolution around a given B-spline, we subdivide the inner points and their triangulated neighbors. To enforce better subdivision, the subdivided knots are tracked to a nearby gradient, using a localized version of $f(\boldsymbol{x}_{j+1}|R, \{\boldsymbol{x}_1, ..., \boldsymbol{x}_j\})$, localized by a Gaussian. The process of subdivision can be performed repeatedly until a sufficiently good resolution is obtained.

The local forces, as well as the effect of smoothing is illustrated in Figure 11.5. In Figure 11.5 (b), the forces are seen to be directed towards the corpus callosum both upwards and downwards. In average they even out, so there is no net force on the parameter. In Figure 11.5(a) the differences are seen to be intact after the smoothing. Therefore the refinement function has an output, and the B-spline is selected for subdivision.

11.5 Evaluation and Results

To support the methodological considerations presented in the previous section, we applied our implemented method on two data sets, both quite different in nature. Our image registration algorithm is implemented for 2D images, due to the 2D nature of our applications, but all observations and equations stated in the current work are equally valid for three dimensional data.

11.5.1 Groupwise Corpus Callosa Registration

To demonstrate the presented method, we did a groupwise registration of 62 mid-sagittal cross-section MR images of the corpus callosum brain structure. To apply the refinement measure (11.13) to a group of images, instead of on a single image, we calculate the sum over the whole data set for each B-spline, in order to ensure that our mesh refinement yields the biggest groupwise improvement in terms of minimization of the sum of squared differences.

This data set is part of the LADIS (Leukoaraiosis and DISability) study [78], a pan-European study involving 12 hospitals and more than 700 patients. Each corpus callosum has been manually segmented by a clinical expert. We used these segmentations for further assessment of the method (see Table 11.1).

All images are registered to one image in the group, and in Figure 11.7 some results of the image registration algorithm are seen. The results of our implementation are compared to the results obtained by rigid alignment and by using an FFD algorithm, which is based on tensor product B-splines. Our quantitative studies showed that the FFD with slightly more control points gave inferior results, but still a significant improvement from the results of the rigid registration alone. The pre-computations of the multivariate splines were a lot more time consuming, but in the group-wise registration, this time was regained in the optimization step, due to the reduced number of control points.

11.5.2 Optical Flow Estimation

Baker et al. recently presented a database for comparison of optical flow results, where the quantitative results of optical flow estimations can be compared to other available algorithms [2]. The optical flow problem resembles the image registration problem a lot, in the sense that we seek to identify correspondences, or flow, across image pairs.

Method	DICE median	Sens.	Spec.	N	Init. time	Run time
Multi.	0.85	0.91	0.99	407	32	149
FFD	0.83	0.91	0.98	640	1	221
Rigid	0.70	0.80	0.98			

Table 11.1: Comparison of the warped segmentations and the reference segmentation after registration using a rigid transformation, an FFD based method, and our multivariate method. Notice how the initialization time (seconds) is larger, but the group-wise running time is smaller for the presented method, due to the reduced number of nodes.

Method	Dimetrodon		Venus		Yosemite	
	Av. End pt.	Avg. Ang.	Av. End pt.	Av. Ang.	Av. End pt.	Avg. Ang.
Current Method	0.20	4.09	0.72	10.74	0.16	3.10
Bruhn et al.	0.43	10.99	0.51	8.73	0.08	1.69
Black and Anandan	0.35	9.26	0.55	7.64	0.15	2.65
Pyramid LK	0.37	10.27	1.03	14.61	0.20	5.22
Zitnick et al.	0.94	15.82	0.85	15.48	0.68	11.09
Mediaplayer^{TM}	0.55	30.10	1.08	11.42	0.47	18.50

Table 11.2: Optical flow evaluation results compared to other contemporary methods

To test the current method on a different application we did the optical flow estimation on the three data sets, from the set, where ground truth flow is available, namely the image pairs named Dimetrodon, Venus and Yosemite. In Figure 11.8 the images are shown with the final grid resolution and the estimated flow field is shown along with the ground truth flow fields.

The results are summarized in Table 11.2. It is seen that our implementation performs significantly better on the one data set than any of the other methods in question, and slightly worse than the best, for the two other data sets.

11.6 Conclusion

The current work has several contributions to the field of parametric image registration.

With an offset in variational optimization theory, we have derived the parametric version of the elastic potential regularization and in effect illustrating how the whole class of differential operator derived regularizers, i.e. curvature and bending energy, can be easily implemented in a parametric setting.

With the same methodology we have analyzed the inherent smoothing or averaging

cost, of selecting warp parameterizations at a specific kernel resolution, in comparison to choosing a finer resolution of the warp kernels. Based on these observations we have proposed a refinement measure, which is shown to be efficient for guiding the local mesh layout.

Though both previous results are useful in their own merits, we have introduced the recently emerged multivariate B-splines based on Delaunay configurations, to the field of image registration. With the combination of our refinement measure and the local flexibility of the multivariate B-splines, we are able to automatically refine the warp field in areas where it results in the minimization of the registration cost function. In effect we get something close to a segmentation of objects allowing for better local control, even where very inhomogeneous areas share a border.

APPENDIX

11.6.1 Implementation

The inverse compositional optimization approach by Baker et al. was used in our implementation to achieve a fast optimization [1]. We obtain a minimum by iteratively minimizing

$$\begin{aligned} \mathcal{J}_{ic}(\boldsymbol{c}) &= \sum_{\boldsymbol{x}} (R(\boldsymbol{\varphi}(\boldsymbol{x}, \Delta\boldsymbol{c})) - I(\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{c}))^2 \\ &\quad + \sum_i F_i^2(\boldsymbol{ct} - \frac{\partial \boldsymbol{c}'}{\partial \Delta \boldsymbol{c}} \Delta \boldsymbol{c}) \end{aligned} \tag{11.18}$$

with respect to $\Delta\boldsymbol{c}$, and the regularizer is expressed as $F_i^2(\boldsymbol{c})$ as derived in Appendix 11.6.2. The parameter updating of $\boldsymbol{c}$ is done according to

$$\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{c}') \leftarrow \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{c}) \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{x}, \Delta\boldsymbol{c}). \tag{11.19}$$

11.6.2 Elastic regularization on Multivariate B-splines

In this section the elastic regularizer and the Lamé operator are derived for the multivariate B-splines, as. To use the inverse compositional algorithm for the image registration, we formulate the regularizer as a sum of functions on the parameters

$$S[\boldsymbol{u}] = \int_\Omega \frac{\lambda}{2} (\nabla \cdot \boldsymbol{u})^2 + \frac{\mu}{4} \sum_{i,j=1}^{s} \left[\frac{\partial \boldsymbol{u}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{u}_j}{\partial \boldsymbol{x}_i} \right]^2 d\, x \tag{11.20}$$

Using the basis representation (11.17) of $\boldsymbol{u}$, we can represent the elastic operator $\mathcal{A} = \mu\Delta\boldsymbol{u} + (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u})$ by the following parameterizations

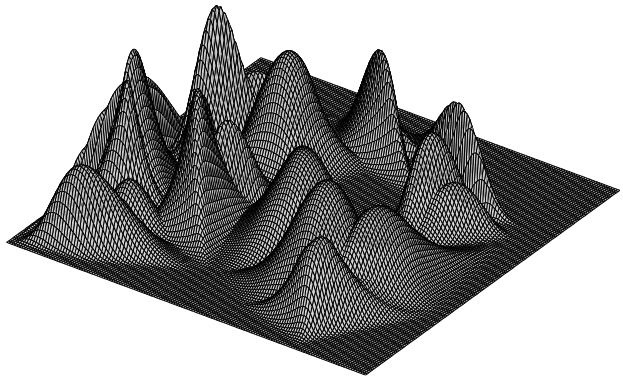
$$\Delta\boldsymbol{u} = \sum_{i=1}^{s}\sum_{I\in\mathcal{I}} \boldsymbol{c}_I \frac{\partial^2}{\partial x_i^2} B_I = \sum_{I\in\mathcal{I}} \boldsymbol{c}_I \sum_{i=1}^{s} \frac{\partial^2}{\partial x_i^2} B_I$$
$$\nabla(\nabla\cdot\boldsymbol{u}) = \sum_{i,j=1}^{s}\sum_{I\in\mathcal{I}} \boldsymbol{e}_i c_{Ij} \frac{\partial^2}{\partial x_i \partial x_j} B_I \tag{11.21}$$

For the inverse compositional optimization the regularization term must be formulated as $\mathcal{S}[\boldsymbol{u}] = \sum_i F_i^2(\boldsymbol{c})$. We parameterize the terms $\nabla\cdot\boldsymbol{u}$ and $\frac{\partial \boldsymbol{u}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{u}_j}{\partial \boldsymbol{x}_i}$ by

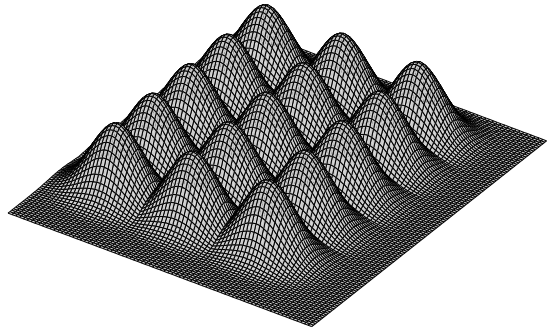
$$\nabla\cdot\boldsymbol{u} = \sum_{I\in\mathcal{I}, j\in\{1,\ldots,s\}} c_{Ij} \frac{\partial}{\partial x_j} B_I \tag{11.22}$$

$$\left[\frac{\partial \boldsymbol{u}_i}{\partial \boldsymbol{x}_j} + \frac{\partial \boldsymbol{u}_j}{\partial \boldsymbol{x}_i}\right] = \sum_{I\in\mathcal{I}} \boldsymbol{c}_{Ii} \frac{\partial}{\partial x_j} B_I + \boldsymbol{c}_{Ij} \frac{\partial}{\partial x_i} B_I \ . \tag{11.23}$$

It is clear that both terms are linear in $\boldsymbol{c}$, which yields the representation of F_i^2, when the integral is discretized.

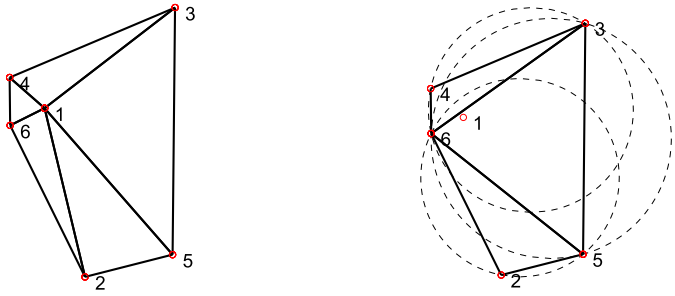


(a) Multivariate B-spline



(b) Tensor product B-spline

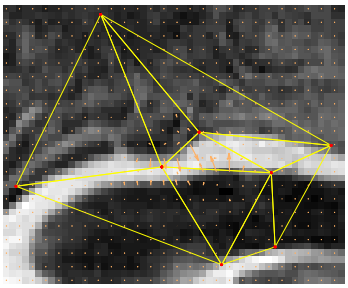
Figure 11.3: Illustration of the flexible kernel of Multivariate B-splines compared to the tensor product B-spline.



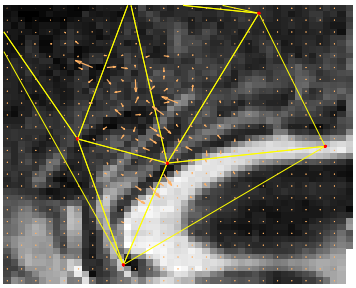
(a) A cell

(b) The Delaunay configurations

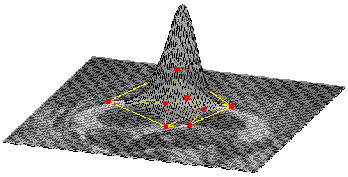
Figure 11.4: Illustration of a cell of a first order Delaunay configuration. Notice how point **1** is inside the 3 triangles circumscribed spheres forming the Delaunay triangulation of its connected points. They constitute the whole cell with **1** as an inside point.



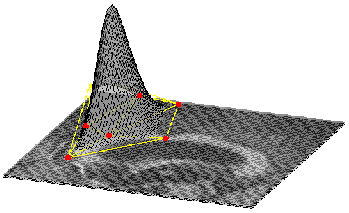
(a) Local 'forces' acting on parameter



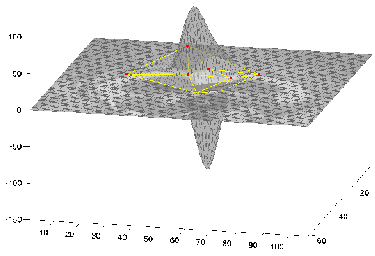
(b) Local 'forces' acting on parameter



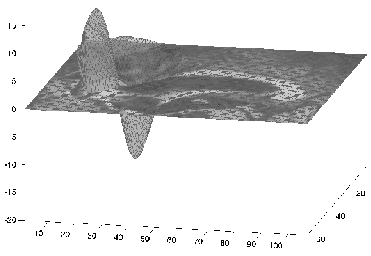
(c) The B-spline basis function



(d) The B-spline basis function

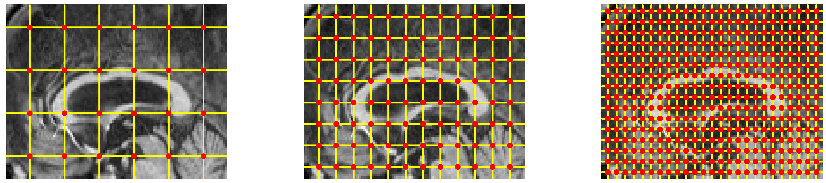


(e) Smoothed projection of forces

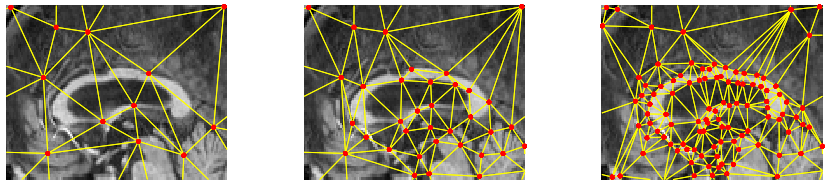


(f) Smoothed projection of forces

Figure 11.5: Illustration of a basis function that will be updated (left column), and one where the impact is not big enough (right column). Notice how (e) and (f) differ by an order of magnitude.



(a) Initial placement (b) 2nd level (c) 3rd level

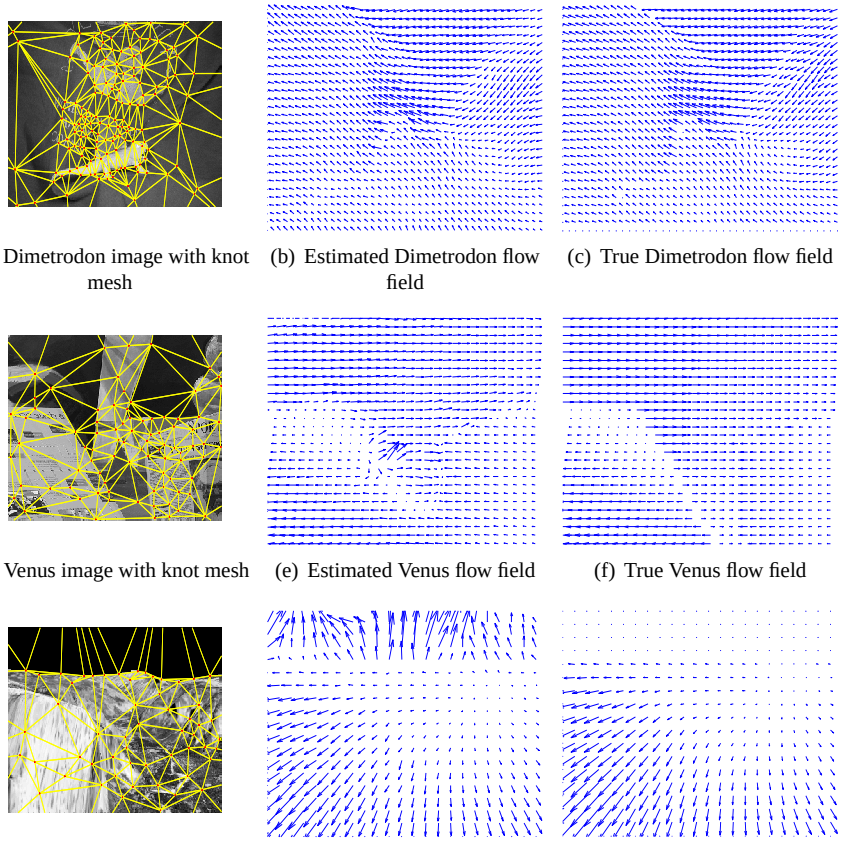


(d) Initial placement (e) 2nd level (f) 3rd level

Figure 11.6: The grid refinement for a uniform setting and our non-uniform setting on the corpus callosum data is illustrated here. Our refinement is based on (11.13). Notice how the structures are nearly segmented by the knots in the non-uniform case.

(a) Rigid (b) FFD (c) Multivariate

Figure 11.7: Checkerboard illustrations of registration results, where the images are tiled from reference and registered image.



(a) Dimetrodon image with knot mesh (b) Estimated Dimetrodon flow field (c) True Dimetrodon flow field

(d) Venus image with knot mesh (e) Estimated Venus flow field (f) True Venus flow field

(g) Yosemite with knot mesh (h) Estimated Yosemite flow field (i) True Yosemite flow field

Figure 11.8: One image of the optical data sets, and the estimated and true flow fields are shown.

CHAPTER 12

Multivariate splines

The current work describes a new explicit formulation of the multivariate simplex splines, based on divided differences and functions resembling the truncated powers, with compact support for efficient evaluation. The formulation is used for deriving practical results for the simplex splines, most notably an explicit formula for the calculation of the derivative of the spline with respect to the position of individual knots. Furthermore, a generalization of Neamtu's spline space is used to define constraints that guarantee the reproduction of polynomials, which is the generalization of the partition of unity to higher order polynomials [76]. Neamtu's results are also used for describing a sub-division scheme for the simplex spline space.

12.1 Introduction

Splines have numerous applications in function approximation, function smoothing, and modelling of form and shape trough free form deformations (FFD). The extension of the curve splines to surfaces and more dimensions was initially approached by de P. de Casteljau and P. Bézier, published by Bézier in 1966, who introduced the Bézier patches [26, 10]. For a better generalization of the univariate splines to more dimensions, Michelli laid some of the foundation for multivariate splines in his work on Kergin interpolation from 1980 [72]. The initial formulation of the multivariate splines were purely geometric, defining the spline as the density of a shadow of a higher dimensional simplex to a lower dimensional simplex, therefore also termed *simplex splines*, e.g. the shadow of an opaque tetrahedron on the two dimensional plane produces a linear simplex spline. Michelli also discovered a recurrence relation for the simplex spline, which facilitated an easier and stable evaluation of the splines based on algebra, rather than geometry. This recurrence relation has since been proved in several different ways [38]. A generalization of univariate truncated powers was introduced, establishing links to the generalization of multivariate divided difference [27, 74, 84]. The cone splines were used for proving more recurrence relations for the multivariate, including a recurrence relation in dimensions, facilitating the calculation of a spline of higher dimension from a set of splines of lower dimension [14].

Dahmen, Michellig and Seidel discussed the construction of spline spaces, and introduced the DMS spline space [72, 90]. Neamtu considered in 2000 the introduction of another more *natural* spline space based on Dealunay configurations [75, 76]. In recent years multivariate splines are also appearing in the computer vision literature [111, 44].

The recurrence relations for calculating B-splines suffers from an explosion in required computations with an increase in dimensionality and order of the splines. The evaluation of an s-dimensional spline of order k requires the evaluation of $(s+1)^k$ simplex splines of order 0 and $((s+1)^k - 1)/s$ barycentric matrices (not all are different). It was argued by Grandine in 1987 that the redundancy could not be efficiently used for optimization, however in 2000 Franssen et al. demonstrated an efficient method for the calculation of B-splines of the DMS space [34, 31]. The current work describes a new explicit formulation of the multivariate simplex splines, based on divided differences and functions resembling the truncated powers, only with compact support for efficient evaluation. The formulation is used for deriving practical results for the simplex splines, most notably an explicit formula for the calculation of the derivative of the spline with respect to the position of individual knots. Furthermore, a generalization of Neamtu's spline space is used to define constraints that guarantee the desirable reproduction of polynomials property [76]. Neamtu's results are also used for describing a sub-division scheme for the simplex spline space.

12.2 Simplex splines

Simplex splines are multivariate generalizations of the univariate B-splines introduced by Curry and Schoenberg [18]. Simplex splines were defined by de Boor in 1976 [23].

Let $\boldsymbol{X} = \{\boldsymbol{x}^0, \ldots, \boldsymbol{x}^{s+k}\}$ be a set of $n = s + k + 1$ points in general position in $\mathbb{R}^s$, then we will refer to $\boldsymbol{X}$ as a set of knots. The simplex spline may be defined as a distribution $M(\boldsymbol{x}|\boldsymbol{X})$, which satisfies the following

$$\int_{\mathbb{R}^s} M(\boldsymbol{x}|\boldsymbol{X}) f(\boldsymbol{x}) d\boldsymbol{x} = s!k! \int_{S^{s+k}} f(\nu_0 \boldsymbol{x}^0 + \cdots + \nu_{s+k} \boldsymbol{x}^{s+k}) d\nu_1 \cdots d\nu_{s+k} \quad (12.1)$$

where

$$S^{s+k} = \{(\nu_0, \ldots, \nu_{s+k}) | \nu_0 + \cdots + \nu_{s+k} = 1, \nu_j \geq 0, j = 0, \ldots, s+k\}$$

Let $V \in X$, the number of points in V is $\#V = s+1$, and let the determinant $d(V) \neq 0$ be nonzero, then it can be shown that $M(\boldsymbol{x}|X)$ satisfies the recurrence relation [38]

$$M(\boldsymbol{x}|X) = \sum_{v \in V} \frac{d({}^{\boldsymbol{x}}_{v} V)}{d(V)} M(\boldsymbol{x}|_v X) \,,$$

where a subscripted ${}_vV$ is the operation of taking out a point, and a superscripted ${}^vV$ indicates the addition of a point and the coefficients $\frac{d({}^{\boldsymbol{x}}_{v} V)}{d(V)}$ are in effect the Barycentric coordinates of $\boldsymbol{x}$ corresponding to the set of points V.

For $k = 0$, ie. $\#X = s + 1$, M is defined as a normalized characteristic function

$$M(\boldsymbol{x}|X) = \frac{\chi_X(\boldsymbol{x})}{\text{vol}_s[X]} = \begin{cases} 1/\text{vol}_s[X] & \boldsymbol{x} \in [X] \\ 0 & \text{otherwise} \end{cases}$$

12.2.1 Divided differences and B-splines

Univariate divided difference is defined by

$$[t_0, \ldots, t_n] f \equiv \sum_{i=0}^{n} \frac{f(t_i)}{\prod_{j \neq i} (t_i - t_j)} \quad (12.2)$$

Truncated powers $(x-t)_+^r$ are defined as $(x-t)^n$ truncated to 0 when $x < t$. B-splines may be defined as the divided difference of a truncated power

$$n[t_0, \ldots, t_n](\cdot - t)_+^n = M(t|t_0, \ldots, t_n) \quad (12.3)$$

where M denotes the univariate B-spline normalized to unit integral. Truncated powers are not very suitable for evaluation due to their non-compact support, however as a theoretical tool they can be used to show that all polynomial splines may be represented by the B-spline basis [22]. Divided difference can be considered an approximation to the derivative operator, and has the B-spline as its Peano kernel [7, 82]. In Section 12.2.2 it is shown how the multivariate version of divided differences may be applied for defining an explicit formula for the simplex spline.

12.2.2 Multivariate divided differences

Let $I = \{I_1, \ldots, I_s\}$ be a set of s indices, $x^I = \{x^{I_1}, \ldots, x^{I_s}\} \in \boldsymbol{X}$, and define $d(x^I, \boldsymbol{x})$ by

$$d(x^I, \boldsymbol{x}) = \begin{vmatrix} x_1^{I_1} & \cdots & x_1^{I_s} & x_1 \\ \vdots & \ddots & \vdots & \vdots \\ x_s^{I_1} & \cdots & x_s^{I_s} & x_s \\ 1 & \cdots & 1 & 1 \end{vmatrix} \tag{12.4}$$

Following the definition presented by Neamtu, we define the multivariate version of the divided difference to be given by [74]

$$[\boldsymbol{x}^0, \ldots, \boldsymbol{x}^n] f = \sum_{I \in \mathcal{N}} \frac{f(x^I)}{\prod_{j \in \mathcal{N} \setminus I} d(x^I, x^j)} \tag{12.5}$$

The following Lemma, which is a straightforward generalization of the univariate case, is needed to prove Theorem 12.2

LEMMA 12.1 *Let $n \geq s$, and $v \in \boldsymbol{X}$, then*

$$[\boldsymbol{X}] d(\cdot, x^j) f(\cdot) = [{}_v\boldsymbol{X}] f(\cdot) \tag{12.6}$$

Neamtu presented a generalization of truncated powers to s dimensions, which allows for multidimensional version of (12.3). However, this representation does not allow for very efficient or stable evaluation of the multivariate B-splines, since it does not have compact support. We introduce a different function, sharing several properties with truncated powers, which is better suited for the purpose of the presented work

$$\hat{T}_{x^j}^{(i)}(x^I, \boldsymbol{x}) = \text{sign}[d(x^I, x^j)] \cdot s! \cdot d^i(x^I, \boldsymbol{x}) \cdot \chi_{I \cup j}(\boldsymbol{x}) \tag{12.7}$$

where $\chi_{I \cup j}(\boldsymbol{x})$ is the characteristic function of the simplex given by the points $\{x^I, x^j\}$.

Define

$$\hat{M}(\boldsymbol{x}|X) = [X]\hat{T}(\cdot, \boldsymbol{x}) \tag{12.8}$$

Then it is clear that for $k = 0$, $\hat{M}$ and M are identical: $\hat{M}(\boldsymbol{x}|X) = \chi_X(\boldsymbol{x})/d(X) = M(\boldsymbol{x}|X)$

THEOREM 12.2 *Let $\hat{M}$ be defined as in (12.8) then the following recurrence relation holds*

$$\hat{M}(\boldsymbol{x}|X) = \sum_{v \in V} \frac{d(\overset{\boldsymbol{x}}{v}V)}{d(V)} \hat{M}(\boldsymbol{x}|_v X) \tag{12.9}$$

PROOF.

$$\begin{aligned}
\sum_{v \in V} \frac{d(\overset{\boldsymbol{x}}{v}V)}{d(V)} \hat{M}(\boldsymbol{x}|_v X) = \sum_{v \in V} \frac{d(\overset{\boldsymbol{x}}{v}V)}{d(V)} [_v X]\hat{T}^{k-1}_{x^j}(\cdot, \boldsymbol{x}) = \\
\sum_{v \in V} \frac{d(\overset{\boldsymbol{x}}{v}V)}{d(V)} [X] d(\cdot, v)\hat{T}^{k-1}_{x^j}(\cdot, \boldsymbol{x}) = \\
[X] \sum_{v \in V} \frac{d(\overset{\boldsymbol{x}}{v}V)}{d(V)} d(\cdot, v)\hat{T}^{k-1}_{x^j}(\cdot, \boldsymbol{x}) = \\
[X] d(\cdot, \boldsymbol{x})\hat{T}^{k-1}_{x^j}(\cdot, \boldsymbol{x}) = [X]T^k_{x^j}(\cdot, \boldsymbol{x}) = \hat{M}(\boldsymbol{x}|X)
\end{aligned} \tag{12.10}$$

It is concluded that $\hat{M}(\boldsymbol{x}|X) = M(\boldsymbol{x}|X)$ since it holds for $k = 0$, and they share the same recurrence relation, so by induction they must be equal for all values $k \in \mathbb{Z}_+ \cup +0$.

12.2.3 Derivatives

Using the formula for the simplex spline introduced in (12.8), the derivative of the simplex spline with respect to the variable $\boldsymbol{x}$ is given by

$$D_{\boldsymbol{x};\boldsymbol{v}} M(\boldsymbol{x}|X) = D_{\boldsymbol{x};\boldsymbol{v}} [X] d^k(\cdot, \boldsymbol{x})\hat{T}^{(0)}_{x^j}(\cdot, \boldsymbol{x}) = k[X] d(\cdot, \boldsymbol{v}) d^{k-1}(\cdot, \boldsymbol{x})\hat{T}^{(0)}_{x^j}(\cdot, \boldsymbol{x})$$

The introduced formula facilitates easy calculation of the derivatives of the simplex spline with respect to the position of the knots, and since nothing is characterizing the

knot $\boldsymbol{x}_j$, it may as well be chosen as the direction of change in the Gateaux derivative

$$D_{\boldsymbol{x}^j;\boldsymbol{v}}M(\boldsymbol{x}|X) = D_{\boldsymbol{x}^j;\boldsymbol{v}}[X]d^k(\cdot,\boldsymbol{x})\hat{T}_j^{(0)}(\cdot,\boldsymbol{x}) =$$
$$D_{\boldsymbol{x}^i;\boldsymbol{v}}s!\sum_{I\in\mathcal{N}}\frac{\text{sign}[d(x^I,x^j)]d^k(I,\boldsymbol{x})\chi_{I\cup j}(\boldsymbol{x})}{\prod_{i\in\mathcal{N}\setminus I}d(x_I,x_i)}) =$$
$$-s!\sum_{I\in\mathcal{N}\setminus\{j\}}\frac{\text{sign}[d(x^I,x^j)]d(I,v)d^k(I,\boldsymbol{x})\chi_{I\cup j}(\boldsymbol{x})}{d(x_I,x_j)\prod_{i\in\mathcal{N}\setminus I}d(x_I,x_i)}\,. \tag{12.11}$$

For normalized simplex splines special care has to be taken, when the knot being altered is also part of defining the normalization coefficient. The derivative of the normalized simplex spline is thus given by

$$D_{\boldsymbol{x}^j;\boldsymbol{v}}N(\boldsymbol{x}|X) = D_{\boldsymbol{x}^j;\boldsymbol{v}}\text{vol}_s[x^B]M(\boldsymbol{x}|X) =$$
$$M(\boldsymbol{x}|X)D_{\boldsymbol{x}^j;\boldsymbol{v}}\text{vol}_s[x^B] + \text{vol}_s[x^B]D_{\boldsymbol{x}^j;\boldsymbol{v}}M(\boldsymbol{x}|X) \tag{12.12}$$

where

$$D_{\boldsymbol{x}^j;\boldsymbol{v}}\text{vol}_s[x^B] = \begin{cases} \frac{\text{sign}[d(x^B)]}{s!}d(x^{jB},v)\,, & j\in B \\ 0 & j\notin B \end{cases} \tag{12.13}$$

12.3 Multivariate B-spline basis

Let $p\in\Pi_k$ be a polynomial of degree k defined in $\mathbb{R}^s$. The polar form $P\in\Pi_k$ is a function of k vector variables $\boldsymbol{x}^1,\ldots,\boldsymbol{x}^k\in\mathbb{R}^s$. It satisfies that $P(\boldsymbol{x},\ldots,\boldsymbol{x}) = p(\boldsymbol{x})$ is symmetric and multi-affine in all the variables $\boldsymbol{x}_i$.

$$p(\boldsymbol{x}) = \sum X\in\Delta_k P(X_I)N(\boldsymbol{x}|X)\,. \tag{12.14}$$

Especially if $p(\boldsymbol{x}) = 1$, $P = 1$, and (12.14) is seen to demonstrate a partition of unity. Liu and Snoyeink generalizes the result of Neamtu [68]. The simplex spline as well as the B-spline by Neamtu is shown to provide a reproduction of polynomials. This reproduction property is satisfied as long as the boundary simplices of Delaunay configurations keep their sign, which is the only condition necessary for the splines to reproduce polynomials. The simplex simplex spline space is spanning the whole B-spline space, and it is less symmetric, which may prove an advantage considering the discussion in Section 10.3.

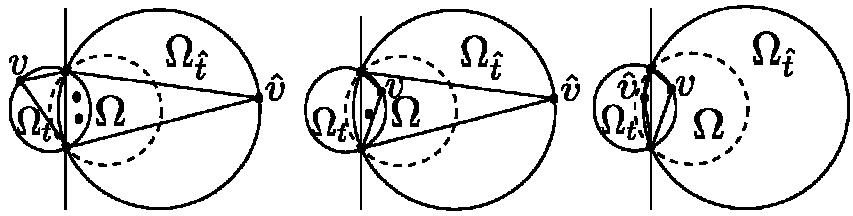
12.3.1 Configurations and facet-matching

Define a boundary-interior configuration as a pair of points (t, I), where t is a tuple of $s+1$ points, forming a simplex, and I is a set of k points disjoint from t. Define a normalized simplex spline by

$$N(\cdot|t, I) \equiv d(t)M(\cdot|t \cup I) \,, \tag{12.15}$$

where we require that t is ordered such that $d(t) > 0$. Neamtu showed that the set of Delaunay configurations of order k defines a basis for reproduction of polynomials of order k. Let Ω_t be the closed ball circumscribed to the simplex defined by the *boundary points* t, then a Delaunay configuration (t, I) is defined as above, where $(I \cup K) \cap \Omega_t = t \cup I$, which means the circumscribed ball, besides t itself, contains only the *interior points* I, from K.

We define faces of a configuration by $F =_v t$, which is essentially a face of the simplex of the boundary points t. For a Delaunay configuration we can define three types of faces, essential, non-essential and phantom faces, as described in Figure 12.1.



(a) Non-essential faces (b) Essential faces (c) phantom faces

Figure 12.1:

(a) non-essential face F, $\#(\{v, \hat{v}\} \cap \Omega) = 0$, and $v, \hat{v}$ are separated by the hyperplane aff$F$

(b) essential face,l $\#(\{v, \hat{v}\} \cap \Omega) = 1$, and $v, \hat{v}$ are on the same side of the hyperplane aff$F$

(c) phantom face $\#(\{v, \hat{v}\} \cap \Omega) = 2$, and $v, \hat{v}$ are separated by the hyperplane aff$F$

This definition was introduced by Neamtu, who also proved that all faces of Delaunay configurations can be labeled as one of these categories [76]. A consequence of this is that all faces of the configurations are either matched by an *opposite* configuration sharing the same face. The configurations are either order k both of them for the essential faces, order $k-1$ for the phantom faces or two faces of order k and order $k-1$

respectively are sharing a face for the essential faces. These properties are important in the proof by Neamtu of the reproduction of polynomials

The proof is based on recursion in degree of polynomial and splines, and it is well known that a zeroth order tesselation forms a partition of unity. The two conditions, satisfied by Delaunay configurations, that facilitates this proof are

$$M(\boldsymbol{x}|X')d(\tfrac{x}{v}({}^{v}F'_B)) + M(\boldsymbol{x}|\hat{X})d(\tfrac{x}{\hat{v}}({}^{\hat{v}}F'_B)) = 0 \tag{12.16}$$

$$d(\tfrac{x}{v}X_B)M(x|F) + d(\tfrac{x}{\hat{v}}\hat{X}_B)M(x|F) = 0 \tag{12.17}$$

They are a consequence of the fact that the points v and $\hat{v}$ are separated by the hyperplane aff$F$, which means that the determinants in the expressions are equal determinants of matrices with switched columns and opposite signs. This observation was also made by Liu and Snoyink, who used it to prove that their link-triangulation algorithm also would form a partition of unity [68]. In the current work, it facilitates the conclusion drawn in Section 12.4.

12.4 Optimizing the approximation

We propose to optimize the approximation using the conclusions from Sections 12.3 and 12.2. One way to optimize the approximation would be fitting the parameters based on gradient information and a fitness function similar to the one presented by Hansen et al. to add knots [44].

12.4.1 Knot movement

It appears favorable to be able to optimize the position of the knots, while retaining the reproduction of polynomials property of the B-splines. Following the conclusions from Section 12.3, this means that none of the boundary points from any configuration order k or smaller should pass the face spanned by the other boundary points of the boundary simplex, if the face is shared with a configuration of the same order. For practical implementation a slight restriction of these criteria is chosen to avoid that any of the determinants of the boundary points change sign. In the two dimensional case this means that no boundary triangle should flip, but no restriction that the configurations have to remain Delaunay order k needs to be imposed as illustrated in Figure 12.2.

The knot movement changes the values of the splines with configurations including the knot. This change may be described by the Gateaux derivative (12.12), and in a linearization, these derivatives may be perceived as additional basis functions, though the optimization with respect to the knots is obviously a non-linear problem.

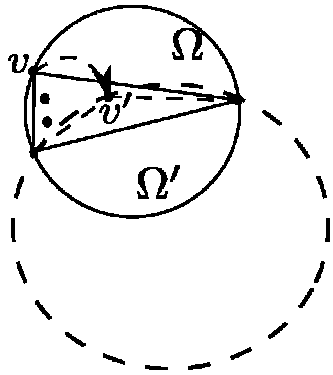


Figure 12.2: Movement of knot v to position v'

12.4.2 Subdivision

If the representation of the simplex splines are insufficient in certain areas as discussed in Section 10.3, it may be an option to add knots to the representation. Adding knots as illustrated in Figure 12.3 will reduce the support of the simplex in $s + 1$ smaller simplices, assuming that the boundary knots are at the boundary. The additionally introduced faces, between the boundary points and the new point v', are all non-essential faces, and by keeping the orientation of the points (by replacing each boundary vertex by the new vertex) the simplex spline may be replaced by the $s + 1$ smaller simplex splines.

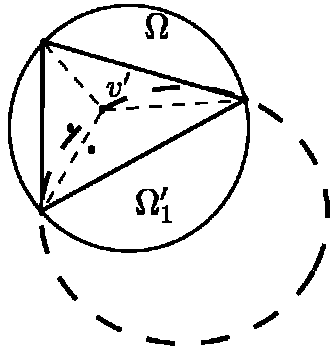
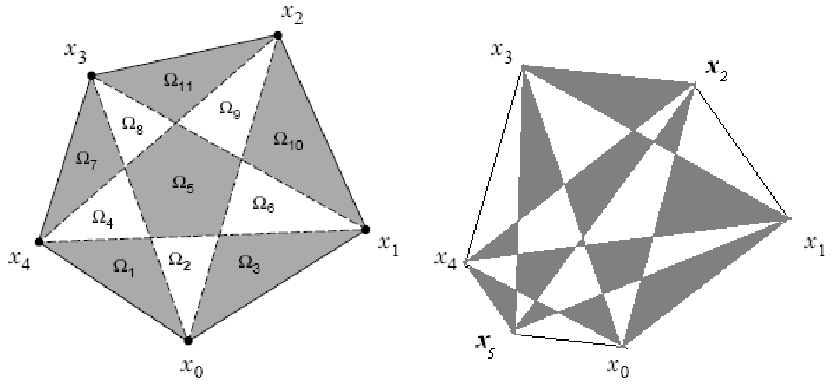


Figure 12.3: Subdivision by adding point v'

12.4.3 Spline evaluation

To evaluate the splines, a number of evaluations of the $\chi_V(\boldsymbol{x})$ have to be performed, which is essentially checking if the point $\boldsymbol{x}$ is inside the primitive defined by the points V. However, this function is clearly piecewise constant, and in Figure 12.4 a two-dimensional simplex is divided in segments where all these functions are constant for any choice of V from $\boldsymbol{X}$. Identifying these segments, and pre-computing the function values and the determinants in (12.5) means a significant speed-up of the spline evaluation. However for cubic simplices, there are typically around 110 segments, and the implementation in 3D, identifying these segments takes around 1 ms runtime, so a significant number of points needs to be evaluated in 3D for the higher order splines, for this be more efficient.



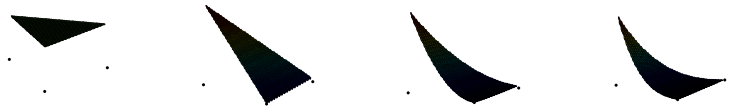
(a) Quadratic simplex (b) Cubic simplex

Figure 12.4: (a) shows the number of segments for a cubic simplex spline, and (b) shows the segments for a cubic spline.

12.5 A view of the splines

The polynomial pieces that are used to piece together the simplex splines in the equation presented in (12.5) are illustrated in Figure 12.5.

In Figure 12.6 a two dimensional slice of a three dimensional set of basis functions are shown. Clearly the basis functions vary significantly in shape and size. In Figure 12.7 a two-dimensional slice through a three-dimensional quadratic simplex spline is illustrated. The derivatives with respect to x, y and z as well as one the knots are



(a) k=0 (b) k=1 (c) k=2 (d) k=3

Figure 12.5: Demonstration of the polynomial pieces composed to form the simplex splines.

illustrated in Figure 12.8, where $\boldsymbol{e}_x$ denote the unit vector in the spatial x direction etc.. In Figure 12.9 the derivatives with respect to one of the knots are shown.

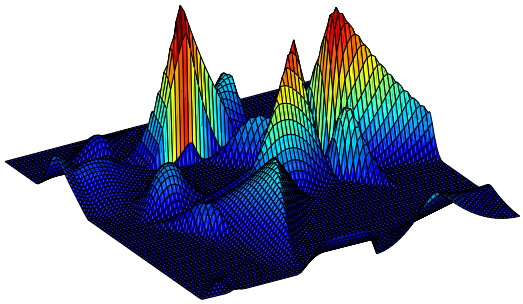


Figure 12.6: Illustration of a slice of some of the Basis functions in a three-dimensional quadratic B-spline basis.

12.6 Conclusion

A new formula, based on the multivariate divided difference, for explicit calculation of the simplex splines has been presented. The formula additionally admits easy calculation of derivatives, both spatial, and with respect to the position of the knots. Though less numerically stable than the recurrence relation, the experiments conducted so far

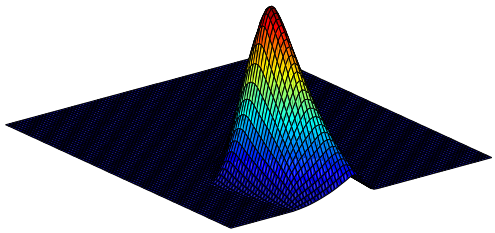
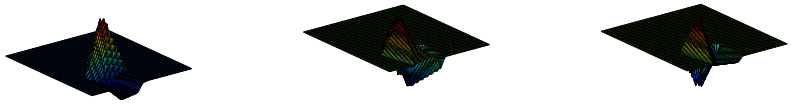
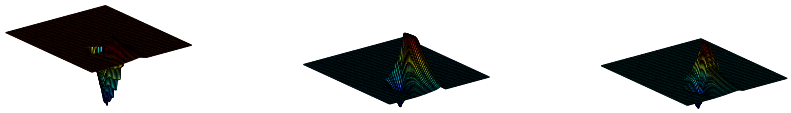


Figure 12.7: Illustration of a slice of a three-dimensional quadratic simplex spline.



(a) $D_{\boldsymbol{x};\boldsymbol{e}_x} M(\boldsymbol{x}|X)$ (b) $D_{\boldsymbol{x};\boldsymbol{e}_y} M(\boldsymbol{x}|X)$ (c) $D_{\boldsymbol{x};\boldsymbol{e}_z} M(\boldsymbol{x}|X)$

Figure 12.8: Illustration of a slice of the spatial derivatives of a three-dimensional quadratic simplex spline.



(a) $D_{\boldsymbol{x}_j;\boldsymbol{e}_x} M(\boldsymbol{x}|X)$ (b) $D_{\boldsymbol{x}_j;\boldsymbol{e}_y} M(\boldsymbol{x}|X)$ (c) $D_{\boldsymbol{x}_j;\boldsymbol{e}_z} M(\boldsymbol{x}|X)$

Figure 12.9: Illustration of a slice of the derivatives of a three-dimensional quadratic simplex spline with respect to change in position for one of the B-spline knots.

indicates that it is well suited for a more efficient calculation of the B-splines, and no numerical instability was observed. The expression may be applied for even faster evaluation of the simplex splines by dividing the simplex into segments that are described by the same polynomials. This is feasible, but for three-dimensional cubic splines, the number of points to be evaluated for a simplex spline has to be in the order of thousands for this approach to be more efficient.

It has been noted that, using simplices inspired by Neamtu, conditions may be set on the knot movements, which ensures that the splines form a partition of unity, even if the knots are not Delaunay. A subdivision scheme is also presented, which requires no recalculation of the configurations of the splines. It is anticipated that the splines will be of great use for specialized problems, where the basis may be tailored to the problem at hand.

CHAPTER 13

Conclusion

The OECD predicts that public health care costs in the member countries will increase from an average of 7 % of the gross domestic products in 2005 to exceed 12 % in 2050. This overgrowth of expenses is partly due to longer life expectancies and survival rates after critical diseases. However, the overgrowth will mainly be caused by implementation of new advanced technology for diagnostic support, monitoring, and treatment. This fact both creates a need for financing health care costs and a huge market for health care technology to be exploited. In order to ensure access to and financing of continued high quality health care in Denmark, emphasis should be put on research, development, and commercialization of health care technology in the world market. The work presented here seek to face several challenges in the analysis of medical image analysis. More precisely it consists of contributions in following three areas.

In Part I an efficient algorithm for calculating the entire regularization path of the support vector domain description is presented. The ability to calculate the entire path with a complexity in the same order as solving the original quadratic problems gives inspiration to utilize the extra information available from the entire path. A method for hierarchical support vector clustering, based on information from the entire regularization path, and multiple Gaussian kernels is described. This method can be considered an extension of a previously developed method, which is also empirically shown to give good results on real world data [40]. Testing the methods on more data would be interesting in the future to draw more conclusions on the nature of the support vector domain description for clustering. However, this would also be very interesting to

see combined with the analysis presented in Chapter 5, where Bayesian methods are applied in the attempt to draw direct statistical conclusions from the support vector domain description analysis.

In Part II different assumptions on the warp fields, namely diffeomorphism and a linear elastic potential in the form of regularization are discussed. Chapter 8 introduces a new warp representation which allows statistical analysis on an unrestricted linear parameter space, where all derivatives are defined. Furthermore, it is shown that L_2-norm on the parameter space introduces a reasonable metric in the actual space of modelled diffeomorphisms. The results compare well to those obtained using Cootes' deformation model. A new parametrization of 3D deformation fields, using potentials and Helmholtz decomposition is presented. The representation can be considered a natural parameterization for both elastic and fluid image registration due to the decoupling of the parameters. For morphometry it is demonstrated that one of the two potentials directly gives us the vorticity of the deformation field. The determinant gradient field is shown to be the first-order small-deformation approximation to the determinant of the Jacobian matrix – probably the most accepted morphometry measure used.

Part III discusses spline approximations of functions, and in particular image registration warp fields. It is shown how spline bases may be learned from the optimization process, ie. image registration optimization, and how this may contribute with a reasonable prior, or regularization in the method. A new formula, based on the multivariate divided difference, for explicit calculation of the simplex splines is presented. The formula additionally admits easy calculation of derivatives, both spatial, and with respect to the position of the knots. Though less numerically stable than the recurrence relation, the experiments conducted so far indicate that it is well suited for a more efficient calculation of the B-splines, and no numerical instability is observed. The expression may be applied for even faster evaluation of the simplex splines by dividing the simplex into segments that are described by the same polynomials. This is feasible, but for three-dimensional cubic splines, the number of points to be evaluated for a simplex spline has to be in the order of thousands for this approach to be more efficient. It is demonstrated, using simplices inspired by Neamtu, that conditions may be set on the knot movements, which ensures that the splines form a partition of unity, even if the knots are not Delaunay. A subdivision scheme is also presented, which requires no recalculation of the configurations of the splines. It is anticipated that the splines will be of great use for specialized problems, where the basis may be tailored to the problem at hand. In Chapter it is demonstrated how the parametric version of the elastic potential and in effect illustrating how the whole class of differential operator derived regularizers, i.e. curvature and bending energy, can be easily implemented in a parametric setting. With the same methodology, the inherent smoothing or averaging cost, of selecting warp parameterizations at a specific kernel resolution, has been analyzed. A refinement measure has been derived, which is shown to be efficient for guiding the local mesh layout. With the combination of our refinement measure and the local flexibility of the multivariate B-splines, the warp field is automatically refined in areas

where it results in the minimization of the registration cost function. The designed basis gives knots that are close to a segmentation of objects allowing for better local control, even where very inhomogeneous areas share a border.

Bibliography

[1] S. Baker and I. Matthews. Lucas-kanade 20 years on A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, 2004.

[2] Simon Baker, Daniel Scharstein, J.P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A database and evaluation methodology for optical flow. In *ICCV*, 2007.

[3] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, Dec 2001.

[4] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. A support vector clustering method. *Proceedings of conference on Advances in Neural Information Processing Systems*, 2001.

[5] Asa Ben-Hur, David Horn, Hava T. Siegelmann, and Vladimir Vapnik. Support vector clustering. *Journal of Machine Learning*, 2:125–137, 2001.

[6] Kristin P. Bennett and Colin Campbell. Support vector machines: Hype or hallelujah. *ACM SIGKDD Explorations*, 2 (2):1–13, 2000.

[7] B.D. Bojanov, H.A. Hakopian, and A.A. Ahakian. *Spline Functions And Multivariate Interpolations*. Springer-Verlag New York, LLC, 1993.

[8] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *COLT' 92: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. New York, NY, USA: ACM Press, pp. 144-152.*, 1992.

[9] C. G. Broyden. The convergence of a class of double-rank minimization algorithms. *Journal of the Institute of Mathematics and its Applications*, 6:76–90, 1970.

[10] P. Bézier. Définition numériqu des courbes et surfaces i. *Automatisme*, XII:625–632, 1966.

[11] Stephane Canu and Alex Smola. Kernel methods and the exponential family. *Neurocomputing*, 69(7-9):714–720, 2006.

[12] Gary E. Christensen, Sarang C. Joshi, and Michael I. Miller. Volumetric transformation of brain anatomy. *IEEE Trans. Med. Imag.*, 16(6):864–877, 1997.

[13] M. K. Chung, K. J. Worsley, T. Paus, C. Cherif, D. L. Collins, J. N. Gieddd, J. L. Rapoport, and A. C. Evans. A unified statistical approach to deformation-based morphometry. *NeuroImage*, 14 (3):595–606, 2001.

[14] Elaine Cohen, Tom Lyche, and R. F. Riesenfeld. Cones and recurrence relations for simplex splines. *Constructive Approximation*, 3:131–141, 1987.

[15] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *5th European Conference on Computer Vision*, 2:484–498, 1998.

[16] T.F. Cootes, C.J. Twinning, and C.J. Taylor. Diffeomorphic statistical shape models. *British Machine Vision Conference*, 1:447–456, 2004.

[17] Tim Cootes, Carole Twining, Vladimir Petrovic, Roy Schestowitz, and Chris J. Taylor. Groupwise construction of appearance models using piece-wise affine deformations. In *Proceedings of British Machine Vision Conference (BMVC)*, volume 2, pages 879–888, 2005.

[18] H. B. Curry and I. J Scoenberg. On pólya frequency functions iv: The fundemental spline functions and their limits. *J. analyse Math.*, 17:71–107, 1966.

[19] W. Dahmen, C.A. Micchelli, and H.-P. Seidel. Blossoming begets B-splines built better by B-patches. In *Math. Comput.*, volume 59, pages 97–115, 1992.

[20] Wolfgang Dahmen. B-splines in analysis, algebra and applications. Technical report, Institut fur Geometrie und Praktische Mathematik, 1997.

[21] C. Davatzikos, M. Vaillant, S. Resnick, J. Prince, S. Letovsky, and R Bryan. A computerized approach for morphological analysis of the corpus callosum. *J. Comput Assist Tomogr*, 20:88–97, 1996.

[22] Carl de Boor. *A practical guide to splines*. Springer-Verlag, Applied Mathematical Sciences , Vol. 27, 2001.

[23] Charles de Boor. Splines as linear combinations of b-splines. *Approximation Theory*, II:1–47, 1976.

[24] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–451, 2004.

[25] L. Eldén. A weighted pseudoinverse, generalized singular values, and constrained least squares problems. *BIT*, 22:487–502, 1982.

[26] Gerald Farin. *Curves and Surfaces for CAGD. A Practical Guide*. Morgan Kaufmann Publishers, 2002.

[27] R. Farwig. Multivariate truncated powers and b-splines with coalescent knots. *SIAM Journal on Numerical Analysis*, 22 (3):592–603, 1985.

[28] P. Thomas Fletcher, Conglin Lu, Stephen M. Pizer, and Sarang Joshi. Prinicpal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Image Analysis*, 23(8):995–1005, 2004.

[29] R. Fletcher. A new approach to variable metric algorithms. *Computer Journal*, 13:317–322, 1970.

[30] D. Forsey and R. Bartels. Hierarchical B-spline refinement. *ACM Tran. Comp. Graph.*, pages 205–212, 1988.

[31] Michael Franssen, Remco C. Veltkamp, and Wieger Wesselink. Efficient evaluation of triangular B-spline surfaces. In *Comp. Aided Geometric Design*, volume 17, pages 863–877, 2000.

[32] Marc G. Genton. Classes of kernels for machine learning: A statistics perspective. *Journal of Machine Learning Research*, 2:299–312, 2001.

[33] D. Goldfarb. A family of variable metric updates derived by variational means. *Mathematics of Computation*, 24:23–26, 1970.

[34] Thomas A. Grandine. The computational cost of simplex spline functions. *SIAM J. Numerical analysis*, 24(4):887–890, 1987.

[35] David Griffith. An introduction to electrodynamics. *Prentice Hall*, 1999.

[36] S. M. Guo, L. C. Chen, and J. S. H. Tsai. A boundary method for outlier detection based on support vector domain description. *Pattern Recogn.*, 42(1):77–83, 2009.

[37] E. Haber and J. Modersitzki. Image registration with a guaranteed displacement regularity. *International Journal on Computer Vision*, 71:361–372, 2007.

[38] Hakop Hakopian. Multivariate spline functions, B-spline basis and polynomial interpolations. *SIAM J. Numerical analysis*, 19(3):510–517, 1982.

[39] M. S. Hansen, H. Ólafsdóttir, K. Sjöstrand, S. G. Erbou, H. B. Larsson, M. B. Stegmann, and R. Larsen. Ischemic segment detection using the support vector domain description. In The International Society for Optical Engineering (SPIE), editor, *International Symposium on Medical Imaging*, feb 2007.

[40] M. S. Hansen, K. Sjöstrand, H. Olafsdóttir, H. B. W. Larsson, M. B. Stegmann, and R. Larsen. Robust pseudo-hierarchical support vector clustering. In *Proceedings of Scandinavian Conference on Image Analysis 2007*, 2007.

[41] Mads Fogtmann Hansen, Michael Sass Hansen, and Rasmus Larsen. Conditional statistical model building. In *International Symposium on Medical Imaging*, 2008.

[42] Michael Sass Hansen, Niels Vorgaard Christensen, and Rasmus Larsen. Curl-gradient image warping : Introducing deformation potentials for medical image registration using helmholtz decomposition. In *Proceedings of International Conference on Computer Vision Theory and Applications*, volume In Press, 2009.

[43] Michael Sass Hansen, Ben Glocker, Rasmus Larsen, and Nassir Navab. Adaptive parameterization of multivariate b-splines for image registration. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[44] Michael Sass Hansen, Benjamin Glocker, Nassir Navab, and Rasmus Larsen. Adaptive parametrization of multivariate b-splines for image registration. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[45] Michael Sass Hansen, Mads Fogtmann Hansen, and Rasmus Larsen. Diffeomorphic statistical deformation models. In *Nonrigid registration and Tracking Through Learning at ICCV*, 2007.

[46] Michael Sass Hansen, David Alberg Holm, Karl Sjöstrand, Karsten Dan Ley, Ian John Rowland, and Rasmus Larsen. Multiscale hierarchical support vector clustering. In *International Symposium on Medical Imaging*, 2008.

[47] Michael Sass Hansen, Thomas Mosbech, Bjarne Ersbøll, and Rasmus Larsen. Estimating local warp field confidence. In *SPIE - International Society for Optical Engineering: International Symposium on Medical Imaging*, 2010.

[48] Michael Sass Hansen, Hildur Olafsdóttir, Tron Andre Darvann, Nuno V. Hermann, Estanio Oubel, Rasmus Larsen, Bjarne Kjær Ersbøll, Alejandro F. Frangi, Per Larsen, Chad A. Perlyn, Gillian M. Morris-Kay, Sven Kreiborg, Miles Wernick, and Jeffrey A.. Fessler. Estimation of independent non-linear deformation modes for analysis of craniofacial malformations in crouzon mice. In *IEEE International Symposium on Biomedical Imaging*, 2007.

[49] Michael Sass Hansen, Karl Sjöstrand, and Rasmus Larsen. On the regularization path of the support vector domain description. *Pattern Recognition Letters*, submitted, 2009.

[50] Michael Sass Hansen, Fei Zhao, Honghai Zhang, Bjarne Kjær Ersbøll, Andreas Wahle, Thomas Scholz, and Milan Sonka. Diagnosis of connective tissue disorders based on independent component analysis of aortic shape and motion from 4d mr images. In *1st International Workshop on Computer Vision for Intravascular and Intracardiac Imaging.*, 2006.

[51] Michael Sass Hansen, Fei Zhao, Honghai Zhang, Nicholas E. Walker, Andreas Wahle, Thomas Scholz, and Milan Sonka. Detection of connective tissue disorder from 3D MR images using independent component analysis. *CVAMIA*, pages 13–24, 2006.

[52] Per Christian Hansen. The discrete Picard condition for discrete ill-posed problems. *BIT*, 30:658–672, 1990.

[53] Per Christian Hansen. The l-curve and its use in the numerical treatment of inverse problems. *Computational Inverse Problems in Electrocardiology*, pages 119–142, 2001.

[54] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–1415, October 2004.

[55] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[56] D. A. Holm, C. D. Ley, L. V. Søgaard, H. J. Simonsen, P. E. Krisjansen, E. L. Lund, and I. J. Rowland. In vivo monitoring of angiogenesis within matrigel chambers using mri. *Proc Intl Soc Magn Reson Med - MR of Cancer Study Group Workshop*, 2006.

[57] H. Hsiao, H. Chen, T. Lin, C. Hsieh, M. Chu, G. Liao, and H. Zhong. A new parametric nonrigid image registration work based on helmholtz's theorem. *SPIE Symposium on Medical Imaging 2008*, 2008.

[58] S. Joshi. Large deformation diffeomorphisms and gaussian random fields for statistical characterization of brain sub-manifolds. *PhD thesis, Sever institute of technology, Washington University*, 1997.

[59] S. Joshi, B. Davis, M. Jomier, and G. Gerig. Unbiased diffeomorphic atlas construction for computational anatomy. *Neuroimage*, 23:151–160, 2004.

[60] Timo Kohlberger, Étienne Mémin, and Christoph Schnörr. Variational dense motion estimation using the helmholtz decomposition. *4th International Conference, Scale Space 2003*, 2695:980, 2003.

[61] M. Kragh, P. J. Hjarnaa, E. Bramm, Paul E.G. Kristjansen, J. Rygaard, and L. Binderup. In vivo chamber angiogenesis assay: an optimized matrigel plug assay for fast assessment of anti-angiogenic activity. *Int. J. Oncol.*, 22:305–311, 2003.

[62] J. Kybic and M. Unser. Fast parametric elastic image registration. *IEEE Transactions on Image Processing*, 12(11):1427–1442, November 2003.

[63] Hildur Ólafsdóttir, Michael Sass Hansen, Karl Sjöstrand, Tron Andre Darvann, Nuno V. Hermann, Estanislao Oubel, Bjarne Kjær Ersbøll, Rasmus Larsen, Alejandro F. Frangi, Per Larsen, Chad A. Perlynn, Gillian M. Morriss-Kay, Sven Kreiborg, Bjarne Kjær Ersbøll, Janne Heikkilä, Ivar Austvoll, and Ingela Nyström. Sparse statistical deformation model for the analysis of craniofacial malformations in the crouzon mouse. In *Proceedings of Scandinavian Conference on Image Analysis*, 2007.

[64] Hildur Ólafsdóttir, Henrik Pedersen, Michael Sass Hansen, Mark Lyksborg, Sune Darkner, and Rasmus Larsen. Registration-based interpolation applied to cardiac MRI. In *Symposium on Medical Imaging*, 2010.

[65] Rasmus Larsen, Karl Sjöstrand, Michael Sass Hansen, and Bjarne Kjær. Ersbøll. Keynote address: Predicting clinical outcome from image derived morphological change. In *Proceedings of 2nd the International Workshop on Image Analysis and In-Vivo Pharmacology*, 2008.

[66] S. Lee, G. Wolberg, K.-Y. Chwa, and S. Y. Shin. Image metamorphosis with scattered feature constraints. *IEEE Trans. Visualizat. Comput. Graphics*, 2(4):337–354, 1996.

[67] Carsten Dan Ley, Lise Vejby Søgaard, Helle Juhl Simonsen, Paul E. Krisjansen, Eva Løbner Lund, and Ian John Rowland. In vivo monitoring of angiogenesis within matrigel chambers using mri. In *International Society for Magnetic Resonance in Medicine, MR of Cancer Study Group Workshop*, 2006.

[68] Yuanxin Liu and Jack Snoeyink. Quadratic and cubic b-splines by generalizing higher-order Voronoi diagrams. In *SCG'07*, 2007.

[69] David J.C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[70] J. MacQueen. Some methods for classification and analysis of multivariate observations. *in Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability*, 1, 1965.

[71] L. Mauerer. Über die mittelwerte einer funktion einer reellen variablen. *Math. Ann.*, 47:263–280, 1896.

[72] CHARLES A. MICCHELLI. A constructive approach to kergin interpolation in r^k: Multivariate b-splines and lagrange interpolation. *Rocky Mountain Journal of Mathematics*, 10 (3):485–494, 1980.

[73] J. Modersitzki. Numerical methods for image registration. *Oxford Uni. Press*, 2004.

[74] Marian Neamtu. Multivariate divided differences i: Basic properties. *SIAM J. Numerical analysis*, 29:1435–1445, 1992.

[75] Marian Neamtu. What is the natural generalization of univariate splines to higher dimensions? In *Math. Methods for Curves and Surfaces: Oslo 2000*, pages 355–392, 2000.

[76] Marian Neamtu. Delaunay configurations and multivariate splines: A generalization of a result of n. b. delaunay. *Trans. of the American Math. Society*, pages 2993–3004, 2007.

[77] M. Opper and O Winther. Gaussian process classification and svm: Mean field results and leave-oneout estimator. *Advances in large margin classifiers Cambridge, MA: MIT Press.*, pages 43–65, 2000.

[78] L. Pantoni, A. M. Basile, G. Pracucci, K. Asplund, J. Bogousslavsky, H. Chabriat, T. Erkinjuntti, F. Fazekas, J. M. Ferro, M. Hennerici, J. O'brien, P. Scheltens, M. C. Visser, L. O. Wahlund, G. Waldemar, A. Wallin, and D. Inzitari. Impact of age-related cerebral white matter changes on the transition to disability - the LADIS study: Rationale, design and methodology. *Neuroepidemiology*, 24(1-2):51–62, 2005.

[79] Jooyoung Park, Daesung Kang, Jongho Kim, James T. Kwok, and Ivor W. Tsang. SVDD-based pattern denoising. *Neural Computation*, 19:1919–1938, 2007.

[80] G. Pólya. Berechnung eines bestimmten integrals. *Math. Ann.*, 74:204–212, 1913.

[81] Massimiliano Pontil and Alessandro Verri. Properties of support vector machines. *Neural Computation*, 10(4):955–974, 1998.

[82] M. J. D. Powell. *Approximation theory and methods*. Cambridge University Press, 1981.

[83] R. Razavi R. Chandrashekara, R. Mohiaddin and D. Rueckert. Nonrigid image registration with subdivision lattices: Application to cardiac mr image analysis. *In proc. of MICCAI*, page in press, 2007.

[84] Christophe Rabut. Multivariate divided differences with simple knots. *SIAM Journal on Numerical Analysis*, 38 (4):1294–1311, 2001.

[85] D. Rueckert, A. F. Frangi, and J. A. Schnabel. Automatic construction of 3D statistical deformation models of the brain using nonrigid registration. *IEEE Transactions on Medical Imaging*, 22(8):1014–25, 2003.

[86] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Nonrigid registration using free-form deformations: application to breast MR images. *IEEE Trans. on Medical Imaging*, 18(8):712–21, 1999.

[87] J. A. Schnabel, D. Rueekert, M. Quist, J. M. Blackall, A. D. Castellano-Smith, T. Hartkens, G. P. Penney, W. A. Hall, H. Liu, C. L. Truwit, F. A. Gerritsen, D. L. G. Hill, and D. J. Hawkes. A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations. *Proc. of MICCAI*, 2208:573–81, 2001.

[88] B. Schölkopf, J.C. Platt, J. Shawe-Taylor, A.J. Smola, and R.C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001.

[89] T. W. Sederberg and S. R. Parry. Free-form deformation of solid geometric models. *Computer Graphics (Proc. SIGGRAPH '86)*, pages 151–160, 1986.

[90] H.-P. Seidel and A.H. Vermeulen. Simplex splines support surprisingly strong symmetric structures and subdivision. *Curves and Surfaces*, II:443–455, 1991.

[91] Jin Seo and Hanseok Ko. Face detection using support vector domain description in color images. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP apos;04).*, 2004.

[92] D. F. Shanno. Conditioning of quasi-newton methods for function minimization. *Mathematics of Computation*, 24:647–656, 1970.

[93] Rodrigo I. Silveira and Marc van Kreveld. Optimal higher order delaunay triangulations of polygons. *Tech. Report*, 2007.

[94] Karl Sjöstrand and Rasmus Larsen. The entire regularization path for the support vector domain description. *Medical Image Computing and Computer-Assisted Intervention, MICCAI 2006, Copenhagen, Denmark*, 2006.

[95] Karl Sjöstrand, Michael S Hansen, Henrik B Larsson, and Rasmus Larsen. A path algorithm for the support vector domain description and its application to medical imaging. *Medical Image Analysis*, 2007.

[96] Alex J. Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14:199–222, 2004.

[97] Peter Sollich. Bayesian methods for support vector machines: Evidence and predictive class probabilities. *Machine Learning*, 46:21–52, 2002.

[98] A. Sommerfeld. Eine besonders anschauliche ableitung des gaussischen fehlergesetzes. *Festschrift Ludvig Boltznabb gewidmet zum 60*, pages 848–859, 1904.

[99] Volker Strassen. Gaussian elimination is not optimal. *Numerical Mathematics*, 13:354–356, 1969.

[100] Colin Studholme and Valerie Cardenas. Population based analysis of directional information in serial deformation tensor morphometry. *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2007*, 4792:311–318, 2007.

[101] David M. J. Tax and Robert P. W. Duin. Support vector domain description. *Pattern Recognition Letters*, 20:1191–1199, 1999.

[102] David M.J. Tax and Robert P.W. Duin. Support vector domain description. *Pattern Recognition Letters*, 20(11-13):1191–1199, 1999.

[103] David M.J. Tax and Robert P.W. Duin. Outliers and data descriptions. 2001.

[104] David M.J. Tax and Robert P.W. Duin. Support vector data description. *Machine Learning*, 54(1):45–66, 2004.

[105] Michael E. Tipping. The relevance vector machine. *Advances in Neural Information Processing Systems*, 12:652/658, 2000.

[106] V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1995.

[107] B. C. Vemuri, J. Yea, b Y. Chen, and C. M. Leonard. Image registration via level-set motion: Applications to atlas-based segmentation. *Medical Image Analysis*, 7(1):1–20, 2003.

[108] M. Vester-Christensen, S. G. Erbou, S. Darkner, and R. Larsen. Accelerated 3D image registration. In *International Symposium on Medical Imaging 2007, San Diego, {CA}*, feb 2007.

[109] Christian Wachinger, Benjamin Glocker, Jochen Zeltner, Nikos Paragios, Nikos Komodakis, Michael Sass Hansen, and Nasir Navab. Deformable mosaicing for whole-body MRI. In *International Conference on Medical Image Computing and Computer Assisted Intervention MICCAI*, 2008.

[110] Defeng Wang, Lin Shib, Daniel S.Yeung, Eric C.C. Tsang, and Pheng Ann Heng. Ellipsoidal support vector clustering for functional mri analysis. *Pattern Recognition*, 40:2685 – 2695, 2007.

[111] Kexiang Wang, Ying He, and Hong Qin. Incorporating rigid structures in non-rigid registration using triangular B-splines. In *Variational, Geometric, and Level Set Methods in Computer Vision*, volume 59, pages 235–246, 2005.

[112] David Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Computation*, pages 1341–1390, 1996.

[113] Z. Xie and G.E. Farin. Image registration using hierarchical B-splines. *IEEE Transactions on Volume Visualization and Computer Graphics*, 10(1):85–94, 2004.

[114] Jianhua Yang, Vladimir Estivill-Castro, and Stephan K. Chalup. Support vector clustering through proximity graph modelling. 2002.

[115] Jiafan Zhang, Qinghua Yan, Yonglin Zhang, and Zhichu Huang. Novel fault class detection based on novelty detection methods. *Intelligent Computing in Signal Processing and Pattern Recognition*, 345:982–987, 2006.

[116] Darko Zikic, Banjamin Glocker, Michael Sass Hansen, Ali Kameme, and Nassir Navab. Construction of statistical shape models from minimal deformations. In *IJ - 2008 MICCAI workshop - Manifolds in Medical Imaging; Metrics, Learning and Beyond*, 2008.

[117] Darko Zikic, Michael Sass Hansen, Benjamin Glocker, Ali Kameme, Rasmus Larsen, and Nassir. Navab. Computing minimal deformations: Application to construction of statistical shape models. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.